VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

DIPLOMOVÁ PRÁCE

2014 Pavel Zatloukal

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Využití autentikačních mechanizmů jednotného přihlášení s podporou biometrie

Authentication Mechanism Single Sign On with Biometric Support

2014 Pavel Zatloukal

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Zadání diplomové práce

Student: **Bc. Pavel Zatloukal**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2601T013 Telekomunikační technika

Téma: Využití autentikačních mechanismů jednotného přihlášení s podporou biometrie
Authentication Mechanism Single Sign On with Biometric Support

Zásady pro vypracování:

Systémy jednotného přihlášení SSO jsou dnes velmi oblíbené jako autentikační mechanismy. Cílem diplomové práce je navrhnout řešení systému SSO s podporou biometrie.
1. Seznámení s problematikou systémů jednotného přihlášení.
2. Autentikace pomocí biometrických prvků
3. Návrh řešení pomocí open-source nástrojů.
4. Ověření fukčnosti v laboratorních podmínkách.

Seznam doporučené odborné literatury:

Kevin Roebuck, *Single Sign-On (Sso): High-Impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*, Tebbo 2011, ISBN-13: 978-1743044957
Michael E. Schuckers, *Computational Methods in Biometric Authentication: Statistical Methods for Performance Evaluation*, Springer 2010, ISBN-13: 978-1849962018
Podle pokynů vedoucího diplomové práce.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Pavel Nevlud**

Datum zadání: 01.09.2013
Datum odevzdání: 07.05.2014

VYSOKÁ ŠKOLA BÁŇSKÁ-TECHNICKÁ UNIVERZITA Fakulta elektrotechniky a informatiky OSTRAVA

doc. Ing. Miroslav Vozňák, Ph.D.
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení

„Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal."

V Ostravě dne 6. 5. 2014

Podpis:.....

Poděkování:

Rád bych poděkoval Ing. Pavlu Nevludovi, vedoucímu diplomové práce, nejen za umožněný stálý přístup do laboratoře N312, ale především za ochotu a odborné vedení, potřebné pro vypracování mé práce.

Abstrakt

Diplomová práce popisuje návrh a řešení zavedení systému jednotného přihlášení do topologie počítačové sítě. Práce se skládá ze dvou částí, a to teoretické a praktické. Teoretická část práce pojednává o možnostech vícefaktorové autentizace, jejího uplatnění a výhodách. Následně je objasněn princip SSO systémů a podrobně popsán protokol Kerberos. Z oblasti biometrie jsou vysvětleny vlastnosti otisku, přístupových zařízení a jejich optimálního zavedení. Praktická část popisuje zavedení čtečky Upek eikon do Linuxového systému a rozbor systémové součásti PAM. Pro navrženou topologii je detailně popsána instalace a konfigurace Kerbera jakožto silného autentizačního systému včetně potřebných prerekvizit, softwarového serveru Apache2, a to vše při uvedení do činnosti s mechanismem jednotného přihlášení. Funkčnost systému Kerberos je následně kombinována s ověřením dle otisků prstů. Z bezpečnostních nároků je realizována podpora komunikace vrstvou SSL. Nakonec proběhla vlastní testování odolnosti biometrické realizace.

Klíčová slova

Jednotné přihlášení, SSO, Linux, biometrie, autentizace, Kerberos, Apache2 server, open-source

Abstract

This thesis describes the design and implementation of a system solution with Single Sign-On in network topology. The thesis consists of two parts: a theoretical and practical. The theoretical part deals with the possibilities of multi-factor authentication , its application and benefits. Consequently, is illustrated the principle of SSO systems and described in detail the Kerberos protocol. In the field of biometrics fingerprint features are explained, access devices and their optimal implementation. The practical part describes the introduction for Upek eikon reader to a Linux system and then analysis the system components PAM. The proposed topology is described in detail to install and configure Kerberos as strong authentication system including prerequisites, Apache2 server software and all while putting in the work with Single Sign-On mechanism. The functionality of the system Kerberos is then combined with verification by fingerprint. Then for security reasons is implemented support with SSL communication layer. Finally, there was perform of testing biometric implementation.

Key words

Single Sign-On, SSO, Linux, biometrics, authentication, Kerberos, Apache2 server, open-source

Seznam použitých symbolů a zkratek

AAA	**Authentication Authorization Accounting Protocol**
AES	**Advanced Encryption Standard**
AS	**Authentication Server**
DES	**Data Encryption Standard**
DoS	**Denial of Service**
DSA	**Digital Signature Algorithm**
EER	**Equal Error Rate**
FRR	**False Rejection Rate**
FAR	**False Acceptation Rate**
HTTP	**Hypertext Transfer Protocol**
HTTPS	**Hypertext Transfer Protocol Secure**
IDEA	**International Data Encryption Algorithm**
KDC	**Key Distribution Center**
LDAP	**Lightweight Directory Access Protocol**
NAT	**Network Address Translation**
NIS	**Network Information Service**
NSS	**Name Service Switch**
PAM	**Pluggable Authentication Modules**
PKI	**Public Key Infrastructure**
RSA	**Rivest Shamir Adleman**
RSH	**Remote Shell**
SHA	**Secure Hash Algorithm**
SCP	**Secure copy**
SSO	**Single Sign-On**
SSL	**Secure Socket Layer**
SSH	**Secure Shell**
TGS	**Tictet Grading Server**
TGT	**Ticket Graning Ticket**
URL	**Uniform Resource Locator**
USB	**Universal Serial Bus**
VPN	**Virtual Private Network**

Obsah

1. Úvod

Při studiu, pracovním procesu, či v osobních aktivitách se člověku přiřadí mnoho přihlašovacích hesel, či těžko zapamatovatelných uživatelských jmen pro různé organizace.

Zbavení těchto nepříjemností by každému usnadnilo „rozumné“ jednotné přihlášení pro všechny potřebné služby. Tato relativně nová metoda přístupu, která má již má ve světě jistě své uplatnění, usnadňuje uživateli orientaci a dopřává mu jistý komfort v pohybu ve virtuálním světě.

Ovšem tato jednotná autentizace, jak z názvu vyplývá, obsahuje jednotný údaj uživatele. Její největší výhoda má potenciální nedostatek v tom, že jednotná data, která se užívají pro více organizací a serverů v sobě nesou určitou nejistotu bezpečnosti (tzv. jednolitosti). Velký důraz je tedy kladen na silný autentizační systém, který tento prvek implementuje.

Z toho důvodu se nabízí otázka, proč nesloučit jednoduchost SSO přihlašování s podporou bezpečných biometrických ověření, identifikující fyzické vlastnosti uživatele. Biometrické prvky jsou v průběhu let neměnné (např. otisk prstu, scan oka) a jedinečné na každém člověku na planetě. Biometrie je metoda, která se rychle uplatňuje nejen v IT světě, organizacích, ale i v domácnostech. Sloučením dvou, nebo více druhů metod a k tomu odpovídajícího bezpečnostního sdílení a zálohování mezi servery či organizace, se dostane velice bezpečného a intuitivního zpříjemnění při vstupu do určitých služeb pro uživatele jako takového.

Jako systém poskytující jednotné přihlášení je zde použit silný autentizační protokol Kerberos verze 5. Byl zde zvolen jak z důvodu jeho open-source implementace, široké podpory platforem, tak i pro jeho silné bezpečnostní mechanizmy.

Server služby, na který se klient hodlá připojit, je zde zastoupen webovým serverem. Webové servery se staly stěžejní součástí světa internetu, marketingu a s ním souvisejících služeb, koordinací a šíření informací uvnitř organizace. Konkrétně se jedná o realizaci pomocí Apache, což je softwarový webový server, s otevřeným kódem pro většinu platforem.

Po osvědčené a funkční navržené topologii je nezbytné pro síť a autentizační systém stanovit druh a mohutnost zabezpečení. Stabilní a bezpečný systém je základ každé organizace, nebo i standardní počítačové sítě. Uložené informace, či citlivá data je z pohledu administrátora nutné opatřit neproniknutelnými bezpečnostními prostředky. Jen vyvážený bezpečný systém, působící na všech různorodých vrstvách a nenáročný k samotným uživatelům uvnitř, je schopen z pohledu perspektivy uspět.

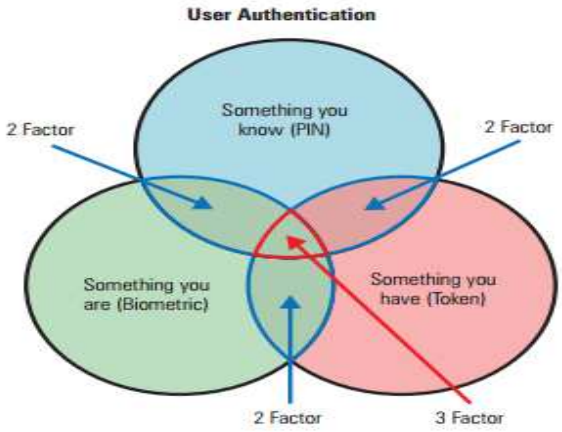
2. Teoretická část

2.1. Vícefaktorová autentizace

V praxi je potřeba chránit citlivé informace a data. K zajištění, aby k datům mohla přistupovat nebo je měnit pouze oprávněná osoba, slouží určité podmínky kontroly informací. Jedná se o proces identifikace, která stanoví, kdo je prověřovaná osoba. Proces ověřování (autentizace) následně potvrzuje nebo popírá identitu dané osoby. Výsledkem procesu je pak povolení nebo zamítnutí přístupu do systému. U autentizace je také vyžadován důkaz identity pro získání jistoty, že osoba je opravdu ta, za kterou se vydává. Existují tři základní způsoby, jak může být osoba ověřena.

První metoda ověřování je založena na něčem, co osoba zná, tzv. znalostní faktor (heslo). Druhá metoda ověřování je založena na něčem, co osoba má, tzv. vlastnický faktor (token). Třetí způsob ověřování je založen na faktu, že jednotlivec sám o sobě vlastní unikátní set měřitelných charakteristik. Tyto charakteristiky, tzv. biometrický faktor, mohou být použity k ověření nebo rozpoznání identity dané osoby (otisk prstu).

Pojem autentizace vyjadřuje proces ověření identity subjektu do systému, databáze, počítače atd. Po dokončení autentizace obvykle následuje autorizace, což je schválení, které umožní povolení přístupu či provedení konkrétní operace subjektem. Proces účtování je sledování činnosti uživatele. Umožňuje sledovat využívání služby ověřeným a autorizovaným uživatelem a zaznamenané výsledky účtování můžou být využity například při fakturaci, nebo při vytváření cílené nabídky pro zákazníka. V oblasti informačních komunikací se o tyto i jiné služby stará AAA protokol. Přihlašování do systémů může být tedy zadáním hesla, technickým řešením, nebo přihlášením pomocí biometrie. Bezpečnost systému zvyšuje zavedení kombinace těchto metod, jak charakterizuje Obr. 1. [8]



Obr. 1: Vícefaktorová autentizace

2.1.1. Autentizace heslem

Jedná se o tzv. „znalostní faktor“. Autentizace za pomocí hesla je pravděpodobně nejstarší a nejčastější metodou. Má ovšem velice mnoho slabých míst, především hrozí snadné vyzrazení hesla. Za použití slovníku běžných slov, který obsahuje 30 000 slov, není problém heslo odhalit. Tento nedostatek je možné částečně eliminovat donucením uživatele použít heslo o minimálním počtu znaku složených z velkých a malých písmen a čísel. Také existuje možnost zaznamenání stisknutých kláves rezidentním programem, který může na stanici běžet bez uživatelova vědomí. Těchto programů je na internetu velké množství a běžný útočník i bez znalosti programování dokáže snadno cizí heslo získat. Z tohoto důvodu byly vyvinuty další spolehlivější metody autentizace.

Za bezpečné heslo můžeme považovat řetězec, který splňuje následující podmínky. Obsahuje velká a malá písmena, speciální a numerické znaky, je dostatečně dlouhé, nemá vztah k uživateli hesla, např. jméno, datum narození, opakující se znaky apod. Proces vytváření bezpečného hesla a dodržování určitých podmínek by měl kontrolovat samotný systém správy hesel. Dále nato by měla být stanovena doba platnosti hesla. Tyto atributy jsou zaznamenány v souboru /etc/passwd.

Username: Password: UserID:GID:UserDescription:HomeDir:Shell
Například:
root: x : 0:0: root: /root/: bin/bash
klient:x:1000:1000:krbklient:/home/klient: /bin/bash

Jméno uživatele (*Username*) – Jméno uživatele většinou zadané malými písmeny. Pro toto pole neexistuje implicitní hodnota.
Zašifrované heslo (*Password*) – Technicky toto pole obsahuje heslo uživatele, ačkoliv zejména v Linuxu jsou používána tzv. stínová hesla, která jsou umístěna v souboru /etc/shadow. Z toho důvodu se v řádku uživatele v souboru /etc/passwd v druhém poli vyskytuje znak x. Tím se informuje instrukce login, že skutečné heslo je uloženo někde jinde.
Identifikační číslo uživatele (*UID*) – Vyjadřuje hodnotu vlastnictví k libovolnému souboru, která je přiřazena uživateli napříč celým systémem.
Implicitní identifikační číslo skupiny (*GID*) – Identifikuje číslo skupiny uživatelů v okamžiku přihlášení. Všechny soubory jsou ve vlastnictví jak uživatele, tak i skupiny.
Popis uživatele (*UserDescription*) – Tento atribut obsahuje popisné informace o uživateli. Může obsahovat například telefonní číslo, poštovní adresu nebo jiné kontaktní informace.
Domovský adresář uživatele (*HomeDirektory*) – Jedná se o definování proměnného prostředí uživatele $HOME. Implicitně ve všech distribucích Linuxu je hodnota domovského adresáře uživatele nastavena na */home/username*.
Uživatelský příkazový procesor (*Shell*) – Když je ověřena totožnost uživatele, přihlašovací program nastaví uživatelskou proměnnou $SHELL na hodnotu z tohoto pole. Implicitně ve všech distribucích Linuxu je příkazovým procesorem nového uživatele Bourne Again Shell, */bin/bash.*

Jelikož Linux používá stínová hesla, jsou zašifrované podoby hesel uloženy v souboru /etc/shadow. Tento soubor obsahuje nejen zašifrované heslo samotné, ale i informace o době platnosti hesla, zda-li není účet zablokován apod.

Username: Password: Lastchange: Minimum: Maximum: Warn: Inactive: Expire
Například:
root: ! : 16063:0: 99999:7:::
klient:boQawhhaCKaxg85c06w:16063:0:99999:7:::

Jméno uživatele (*Username*) – přihlašovací jméno uživatele, které odpovídá položce, která je použita v souboru /etc/passwd.
Zašifrované heslo (*Password*) – v této položce je uložena zašifrovaná podoba skutečného uživatelského hesla, tzv.stínové heslo.
Poslední změna hesla (*Lastchange*) **–** specifikuje počet dní od 1. ledna 1970, kdy byla provedena poslední platná změna hesla.
Minimum na změnu hesla (*Minimum*) – reprezentuje počet dní, po jejichž uplynutí může být heslo změněno. Obvykle je tato hodnota nastavena na 0, což umožňuje uživateli měnit heslo, jak často potřebuje.
Maximum na změnu hesla (*Maximum*) – reprezentuje počet dní, po jejichž uplynutí musí uživatel heslo změnit. Pokud změny hesla nejsou vynuceny, potom tato položka obsahuje hodnotu 99999.
Varování, že u hesla vyprší životnost (*Warn*) – počet dní před lhůtou, než je uživatel varován, že heslo ztratí svoji platnost. Obvykle je uživatel varován jeden týden před vypršením platnosti hesla, proto tato položka obsahuje hodnotu 7.
Počet dní mezi skončením platnosti hesla a zablokováním účtu (*Inactive*) – tato položka reprezentuje počet dní mezi vypršením platnosti hesla a zablokováním uživatelského účtu. Pokud automatické zablokování není požadováno, položka je nastavena na hodnotu -1 nebo je prázdná.
Zablokování účtu – tato položka uvádí počet dní od 1. ledna 1970, kdy byl uživatelský účet zablokován. Když není tento automatický způsob zablokování účtu použit, položka je nastavena na -1 nebo je prázdná.
Speciální příznak (rezerva) – tato položka je rezervována pro budoucí použití. Obvykle je prázdná. [30]

2.1.2. Autentizace technickým řešením

Jedná se o tzv. „vlastnický faktor“. Čipové karty a kryptografické tokeny spadají do oblasti hardwarových kryptografických zařízení. Přesněji spadají do množiny hardwarových kryptografických modulů, v tomto případě označovaných jako autentizace dle předmětů, které slouží k zabezpečenému přenosu a uložení tajných údajů.

Mezi tajné údaje často patří především soukromé klíče. Běžně se však ukládají na hardwarové kryptografické moduly i veřejné klíče v podobě digitálních certifikátů podepsaných certifikační autoritou.

Oba dva druhy klíčů tvoří základ pro asymetrické šifrování, které se používají v infrastruktuře správy a distribuce veřejných klíčů PKI (Public Key Infrastructure). Výhodou hardwarových kryptografických modulů jsou především nesnadné možnosti získání těchto tajných údajů obsažených v jejich paměťových částech. Toho je dosaženo pomoci vlastního kryptografického procesoru, který provádí okamžité šifrování a dešifrování a je umístěn přímo v modulu.

Obr. 2: Autentizační tokeny v praxi [37]

Čipové karty jsou standardizovány a využívají se ve spojení se čtečkami. Nutnost dodatečného připojení čtečky čipových karet k počítači představuje mírnou nevýhodu z pohledu potřeby dalšího periferního zařízení.

Kryptografické tokeny mají oproti čipovým kartám menší rozměry a mezi jejich výhody patří především komunikační rozhraní. To představuje USB port nacházející se u všech současných moderních počítačů.

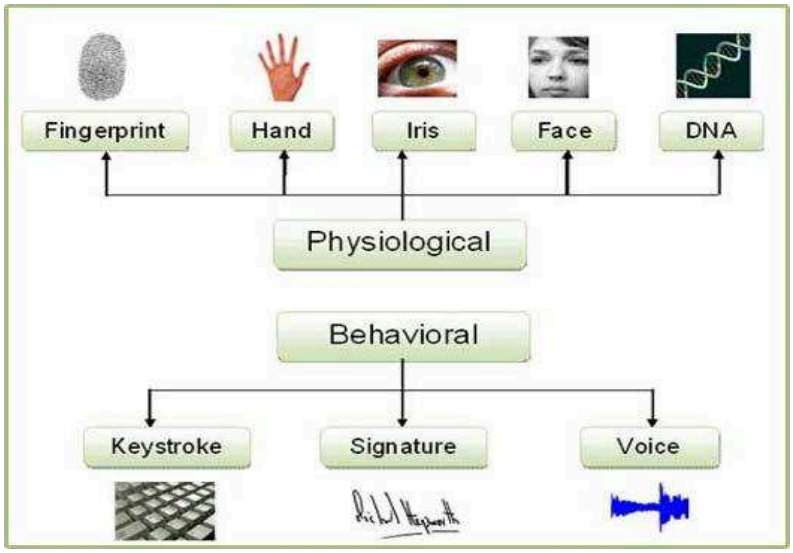
Jakožto vlastnická autentizace, obsahující výhody popsané výše, obsahuje podstatnou nevýhodu v situaci ztráty, nebo cíleného odcizení. [31]

2.1.3. Biometrická autentizace

Jedná se o tzv. „biometrický faktor“. Biometrie se skládá z řeckých slov “bios“ - živý, a “metria“ - měření. Je to metoda autentizace založená na rozpoznávání jedinečných biologických charakteristikách subjektu. Biometrie je souhrn výpočetních technik, které dovolují automaticky rozpoznat jakoukoliv osobu na základě jejich fyzických parametrů. Biometrika se věnuje studiu metod sloužících k rozpoznávání člověka na základě jeho biologických parametrů nebo behaviorálních vlastností. [3]

Mezi možnosti biometrické autentizace patří sítnice oka, duhovka, verifikace pomocí povrchové topografie rohovky, biometrie ušního boltce (podle morfometrických vztahů, termografu ušního boltce, otisku struktur ušního boltce, ozvěny vracené kanálkem), geometrie obličeje, termografu obličeje, geometrie ruky, struktury žil na ruce, verifikace podle tvaru článku prstu a pěstí, verifikace podle vrásnění článků prstů, identifikace podle podélného rýhování nehtů, identifikace podle otisků prstů, identifikace pomocí spektroskopie kůže, identifikace podle pachu, verifikace podle DNA atd. [5, 6]

Autentizace je také možná pomocí tzv. behaviorálních metod. Jsou to unikátní vlastnosti jedince (návyky které nelze napodobit) jako například identifikace podle charakteristiky hlasu, verifikace podle způsobu pohybu očí, verifikace podle tvaru a pohybu rtů, či dynamika stisku kláves. V procesu inicializace je nutné provést sejmutí, získaní referenčního vzorku a odfiltrování nežádoucích jevů, které by mohly požadovaný výsledek před uložením do databáze referenčních vzorků zkreslit.



Obr. 3: Typy biometrických ověření [38]

Pod pojmem vícefaktorová autentizace se pokládá kombinace dvou nebo více faktorů (metod) autentizace. Kombinace více metod autentizace samozřejmě přináší větší míru spolehlivosti a úrovně zabezpečení. Je důležité vhodně zkombinovat jednotlivé metody tak, aby se vzájemně doplňovaly.

Použití metody s heslem je mnohdy nejčastější řešení z důvodu jednoduchosti a levného nasazení této metody, a to za předpokladu, že uživatel použije dostatečně silné heslo, i dostačující úroveň zabezpečení. Skupina technických řešení se běžně využívá při zabezpečení budov v systémech EZS (elektrický zabezpečovací systém), kde nahrazují klasické klíče. Tato metoda je mnohdy drahá a velmi snadno ji lze zneužít při ztrátě nebo krádeži těchto zařízení. Typickým příkladem ze skupiny biometrických řešení může být snímač otisků prstů, který se hojně využívá i v levnějších počítačových sestavách nebo noteboocích.

Vybrat ale vhodnou autentizační metodu není vůbec jednoduché. Zde neplatí čím více, tím lépe. Vždy je třeba znát účel a podmínky, v jakých bude autentizace nasazena. Mezi důležitá kritéria se musí zohlednit kdo, jak často a kolik uživatelů se bude autentizovat. Budou-li mít vzdálený přístup, jaké náklady budou na pořízení a provoz a jaká bude finální bezpečnost.

Následující část práce se zabývá kombinací metody bezpečného jednotného hesla a otisku prstu. Jedná se o mnohdy dostupnou kombinaci metod vhodných pro uživatele, a to i pomocí opensource řešení. [39, 40]

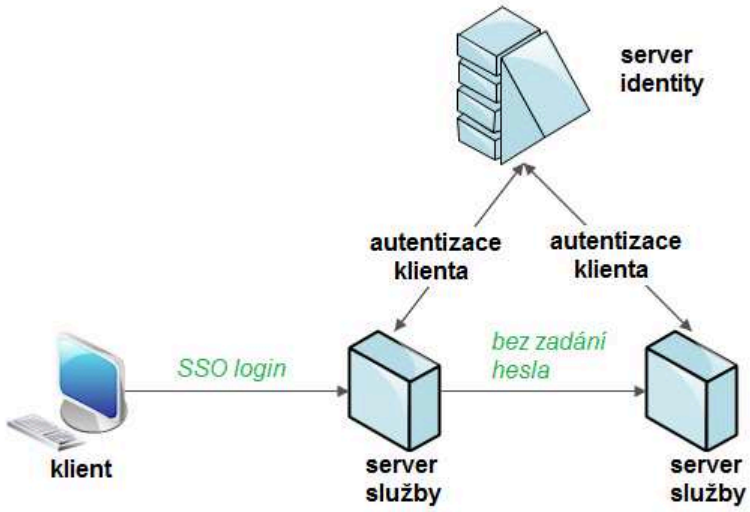
2.1. Jednotné přihlášení SSO

2.1.1. Autentizační systémy

Pojmem Single Sign-On je označena tzv. metoda jednotného přihlášení. Jedná se o autentizační proces umožňující uživateli přistoupit k více aplikacím a zdrojům v rámci jedné relace, kdy je využito možnosti zadání jména a hesla případně jiných identifikátorů pouze jednou. Podstatou technologie SSO je, že namísto velkého množství přihlašovacích atributů (přihlašovací jméno a heslo) si uživatel musí pamatovat atribut jen jeden. Ostatní si za něj pamatuje systém SSO. Soulad se všemi politikami a požadavky tedy není dotčen. SSO se může stát branou do systémů násobných, spojených, ale i zcela nezávislých.

Proces SSO, vyžádaný při zahájení relace, autentizuje uživatele ke všem aplikacím, ke kterým má na serveru vydána přístupová práva a tím eliminuje veškeré následné výzvy k autentizaci. Výzvy k autentizaci by bez zavedeného procesu SSO byly aktivovány vždy, když by se uživatel v rámci relace přepojil mezi aplikacemi. Hlavní cíle zavedení SSO jsou obvykle spojeny s pohodlím, jež SSO přináší celé organizaci a předně uživateli samotnému.

Přínos se zavedením SSO umožňuje předcházení ztráty času spojeného se správou hesel v celém jejich cyklu vytváření uživatelských přístupů do jednotlivých aplikací a obnovování celé sady zapomenutých hesel. [9]



Obr. 4: Hlavní myšlenka jednotného přihlášení (SSO)

V oblasti SSO je mnoho různorodých implementací. Jedná se o systémy, které jsou komerční nebo volně dostupné. Komerční řešení jednotných přihlášení skýtá mnoho výhod, ať se jedná o podporu různých množství podporovaných platforem, či široké možnosti implementace. Oproti tomu volný software, neboli opensource řešení, je sice finančně nenáročné, ovšem podpora platforem již není tak široká, přehledná a rychle implementovatelná. Jednotlivé verze se také liší svým uplatněním a rozsahem.

Typů systémů jednotného přihlášení je mnoho druhů. Mezi významné patří systémy typu Enterprise Single Sign-On. Zabezpečují jednotné přihlášení systémem, jenž po prvotním přihlášení uživatele do primárního systému zachytává následné výzvy sekundárních aplikací k zadání ověřovacích informací a automaticky doplňuje požadované ID a případně heslo.

Dalším typem je webové SSO, někdy nazýváno Web access management (Web-SSO). Umožňuje koncovým uživatelům přístup k aplikacím a zdrojům, které jsou přístupné pomocí internetového prohlížeče. Autentizace je dosaženo pomocí prezentování a archivace identifikačních údajů v cookies, což jsou malá množství dat v protokolu http, která pošle webovému prohlížeči dotazovaný server. Prohlížeč může tato data uložit do počítače uživatele, na proxy serveru, případně na cílovém internetovém serveru. Informace uložená v cookie je použita vždy, když si koncový uživatel vyžádá přístup k webovému portálu nebo novému zdroji dosažitelného pomocí web prohlížeče. Mezi webové SSO autentizační nástroje patří například projekt MojeID, OpenID, Shibboleth apod. Projekt Shibboleth je podporovaný zejména mezi univerzitami a soukromými federacemi, ovšem řešení mnohdy vyžaduje velmi náročnou konfiguraci.

Oproti tomu systém Kerberos byl navržen jako model klient server podporující oboustrannou autentizaci. Oboustrannou autentizací je myšleno vzájemné ověření identity koncových uživatelů a služeb. Koncoví uživatelé se ke Kerberos serveru přihlašují pomocí jejich hesla. Výměnou za heslo získají uživatelé od serveru lístek, kterým klient prezentuje svůj přístup ke službě. Varianta Kerberos je používána jako výchozí autetizační metoda pro Windows 2000, Windows XP a Windows Server 2003.

Dalším přístupem vhodným také pro Web aplikace je Federace identity. Federovaná identita používá standardy protokolů SAML a WEBSecurity k oprávnění potvrzení identity uživatele mezi aplikacemi, čímž zabraňuje nadbytečné potřebě autentizací. Tzv. federování identity umožňuje organizacím poskytovat SSO mezi různorodými sítěmi, při slučování účtů, sítí a aplikací, a k prospěchu zaměstnanců, zákazníků a partnerů.

Základem OpenID je distribuovaný a decentralizovaný proces. OpenID je mechanismem, který váže identitu uživatelů na snadno zpracovatelné URL, které je možné ověřit libovolným serverem, na kterém je protokol spuštěn. V sítích, ve kterých je OpenID povoleno, nepotřebují uživatelé před získáním přístupu ke každé síti vytvářet a spravovat nový účet. Pro autentizaci uživatele do důvěryhodné sítě podporující OpenID je identita uživatele potvrzena a předána dalším sítím podporujícím OpenID. Jelikož je filozofie OpenID odlišná od SSO, nelze spolehnout na důvěryhodnost mechanismu její autentizace, není OpenID použitelné v citlivých oblastech, jakými jsou bankovnictví a on-line nákupy.

Zvolení vhodného systému z výčtu možností SSO implementací záleží již na konkrétním záměru administrátora, federace, jejich prostředků a dle uživatelských nároků. [12, 42]

2.1.1. Kerberos protokol

Protokol Kerberos je centralizovaný autentizační systém, definován v RFC 1510, používající silnou kryptografii pro bezpečné ověření klienta a serveru přes nezabezpečenou síť. Funguje na principu šifrované komunikace, kde základní princip spočívá ve verifikaci dvou uzlů vůči sobě, prostřednictvím tzv. důvěryhodné třetí strany. Centrální autentizační prvek zvyšuje bezpečnost a může poskytovat služby více aplikacím. Mezi jeho výhody patří centralizovaná správa přihlašovacích informací, jednorázové zadání hesla pro všechny služby, snadné začlenění nové služby do existujícího systému, autentizace uživatele vůči službě, ale i služby vůči uživateli. Podstatný je i hlavní klad Kerbera, jenž neposílá hesla po síti, jelikož se klient prokazuje díky lístkům s omezenou časovou platností. Další implementaci Kerbera najdeme u Microsoftu, kde se od MS Windows 2000 používá jako autentizační mechanizmus v Active Directory. Mezi základní prvky Kerbera patří:

Key Distribution Center (KDC) - hlavní části systému Kerberos. Je tzv. důvěryhodnou třetí stranou umožňující bezpečnou autentizaci a autorizaci přístupu ke zdrojům, která bezpečně uchovává data o uživatelích a službách v síti. KDC se dělí na dva subsystémy:

- Autentizační server (**AS**) je odpovědný za autentizaci a autorizaci uživatelů a služeb
- Ticket Grantig Server (**TGS**) přiděluje oprávnění (lístky) k použití služby

Realm - doména (oblast) spravovaná jedním KDC.
Principál – skládá se ze jména (primary) uživatele nebo služby, instance a domény (realm). Principál má tvar primary/instance@REALM.
Lístky (tickets, tikety) - vydává KDC a používají se pro prokazování totožnosti klienta vůči službě, ale i naopak.
Služba - zahrnuje atribut "host" (použití telnet, rsh, ssh), "ftp" (FTP), "krbtgt" (ověřování, udělování lístků), "HTTP" (webový server) a "pop" (e-mail).
Keytab - jsou soubory extrahované z hlavní databáze KDC a obsahují šifrovací klíče pro služby nebo hosty. [19]

Obr. 5: Logo MIT Kerberos verze 5 [10]

KDC se skládá z AS a TGS. Tyto dvě služby jsou samostatné, ale většinou se provozují na společném počítači. Tento počítač by měl být velmi dobře zabezpečený, protože při jeho kompromitaci může útočník získat přístup do celé sítě. Lístky přidělené uživateli mají omezenou platnost, která jde v případě potřeby prodlužovat. Doba se většinou pohybuje v rozmezí 8 až 10 hodin. Po propadnutí lístku si může uživatel požádat o lístek nový. [43, 19]

Kerberos je založen na symetrickém šifrování, to znamená, že dva komunikující uzly mezi sebou sdílí jeden tajný klíč. Právě pomocí tohoto klíče jsou data šifrována i dešifrována. Kerberos neřeší otázku distribuce toho klíče mezi tzv. principály, avšak nabízí prostředky, jak tyto klíče generovat. Pro vytvoření privátního klíče se využívá jednocestná hashovací funkce tzv. „string2key“, díky níž jsou klíče uložené v databázi zpětně nerozluštitelné. V současnosti se nejčastěji využívá AES šifrovací algoritmus. Takto vygenerovaný klíč se řadí do kategorie tzv. dlouhodobých klíčů (long-live key) a nepředpokládá se, že by měl být často pozměňován. V případě uživatele se tento klíč nejčastěji generuje z jeho hesla. V síti využívající Kerberos jsou privátní klíče sdíleny mezi principály a KDC. [36]

Pro pochopení protokolu Kerberos je nutná teoretická znalost průběhu ověření hosta, vyjednání lístku a bezpečnostní postupy. Autentizace klienta dle politiky Kerbera verze 5 probíhá podle následujícího vyjednávání. Prvně je ze strany klienta po vygenerování klíče zaslána zpráva AS_REQUEST na KDC. Tímto klient žádá AS o udělení lístku. Zpráva poslaná KDC je rovněž prvně nešifrovaná a obsahuje:

- Principál klienta
- Principál TGS (krbtgt)
- Klientskou časovou hodnotu (timestamp)
- Požadovanou dobu platnosti lístku - lifetime (obvykle 8 to 10 hodin)

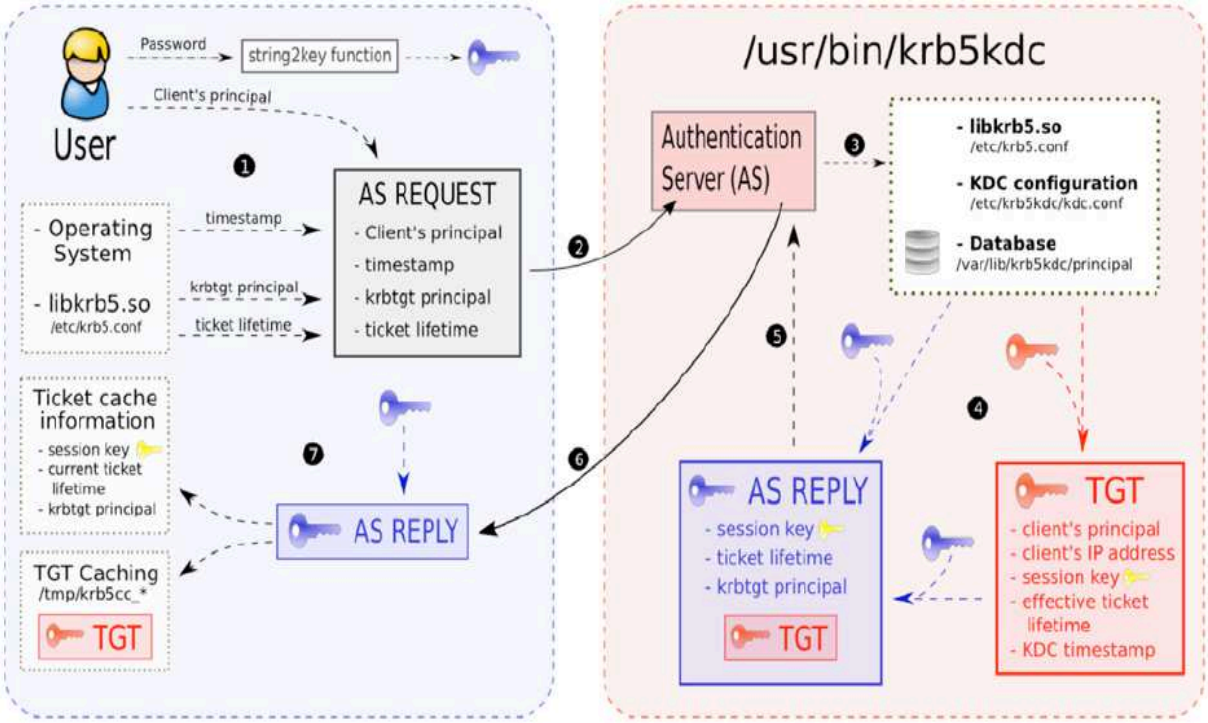
KDC přijme zprávu, zkontroluje klienta v databázi a zkontroluje časovou synchronizaci. Pokud je pre-autentizace povinná, KDC nevrátí TGT před autentizací klienta požadovanou formou. Nejčastěji se jedná o formu PA-ENC-TIMESTAMP, kde je současný timestamp zašifrován klientovým klíčem (na klientské straně získané z hesla funkcí string2key).
V tomto případě je klientem opět zaslána žádost AS_REQUEST, tentokrát v šifrované podobě, s časovým razítkem. Pokud pre-autentizace proběhne úspěšně, dostane klient lístek TGT od KDC ve zprávě AS_REPLY.

Při ověřování server vygeneruje náhodný klíč relace ("krátkodobý" klíč). KDC udělá kopii pro klienta. Ten je přidán do AS_REPLY zprávy. Druhá kopie zůstává k dispozici pro TGS. Tento klíč se používá především pro pozdější kerberizované služby. Za předpokladu, že klient uspěl v jeho ověřování, KDC vrátí zprávu AS_REPLY, obsahující TGT uložený v nějaké formě v cache paměti. Zpráva je šifrována pomocí klíče uživatele. AS_REPLY zpráva je vytvořena ze dvou vrstev, přičemž první z nich je zašifrována pomocí klíče uživatele. Zatímco druhá vrstva je sama o sobě TGT nejprve zašifrována pomocí klíče TGS a znovu následně zašifrována pomocí klíče uživatele. Díky tomu může pouze důvěryhodný (autentizovaný) uživatel dešifrovat zprávy a získat TGT. Zpráva AS_REPLY obsahuje dvojici atributů:

- Zašifrované pomocí TGS klíčem: Kopii klíče relace
  Skutečnou dobu platnosti lístku
  KDC timestamp
  Principal klienta
  IP adresu klienta

- Zašifrované klíčem uživatele: Kopii klíče relace pro uživatele
  Životnost lístku
  Krbtgt principál

I když TGT je dešifrován a uložen v cache paměti na straně klienta, jeho obsah není možné číst dál na straně klienta. Zašifrován je pomocí klíče TGS, který je znám pouze na Ticket Grading Serveru. V této chvíli již klient vlastní TGT a je tedy ověřen u AS.



Obr. 6: Průběh vyjednávání k získání lístku pro uživatele [10]

Následuje proces pro udělení přístupu k požadované službě. Tento mechanizmus požaduje po klientu autentizaci v určité podobě pro následný přístup ke kerberizované službě. Tento předpoklad vyžaduje samostatnou komunikaci s TGS, tzv. zprávou TGS_REQUEST. Zpráva zaslaná klientem se skládá z několika atributů:

- Žádost samotného TGS obsahující servis principál a požadovaný lifetime
- TGT získané dříve (při úspěšné autentizaci)
- Autentikátor

Autentikátor je zde ke zmaření reply odpovědi. To je zpráva zašifrována pomocí klíče relace získané v průběhu procesu a obsahuje principál uživatele a timestamp. Touto cestou KDC zajišťuje, že tato jedinečná zpráva přichází z pravého uživatele. To zjistí nejprve kontrolou dočasného klíče relace vyjednaného dříve. Zadruhé prostřednictvím časového razítka, který detekuje podvodný pokus o reply. Po úspěšné žádosti (platný TGT a regulérní autentikátor) TGS vygeneruje atributy, které budou zaslány zpět klientovi.

V další fázi server generuje novou sadu klíčů relace. Reply zpráva od serveru je šifrována pomocí klíče relace získané při AS procesu. Tedy pouze klient, který se předtím identifikoval na KDC je schopen číst jeho obsah a extrahovat z něj TGS. Tato zpráva tvoří součást TGS_REPLY:

Zašifrování pomocí klíče relace získaným při AS procesu obsahuje:

- Kopii nového klíče relace pro uživatele
- Efektivní lifetime
- Principál služby

Prvně zašifrováno s dlouhodobým klíčem služby TGS a poté s aktuálním klíčem relace obsahuje:

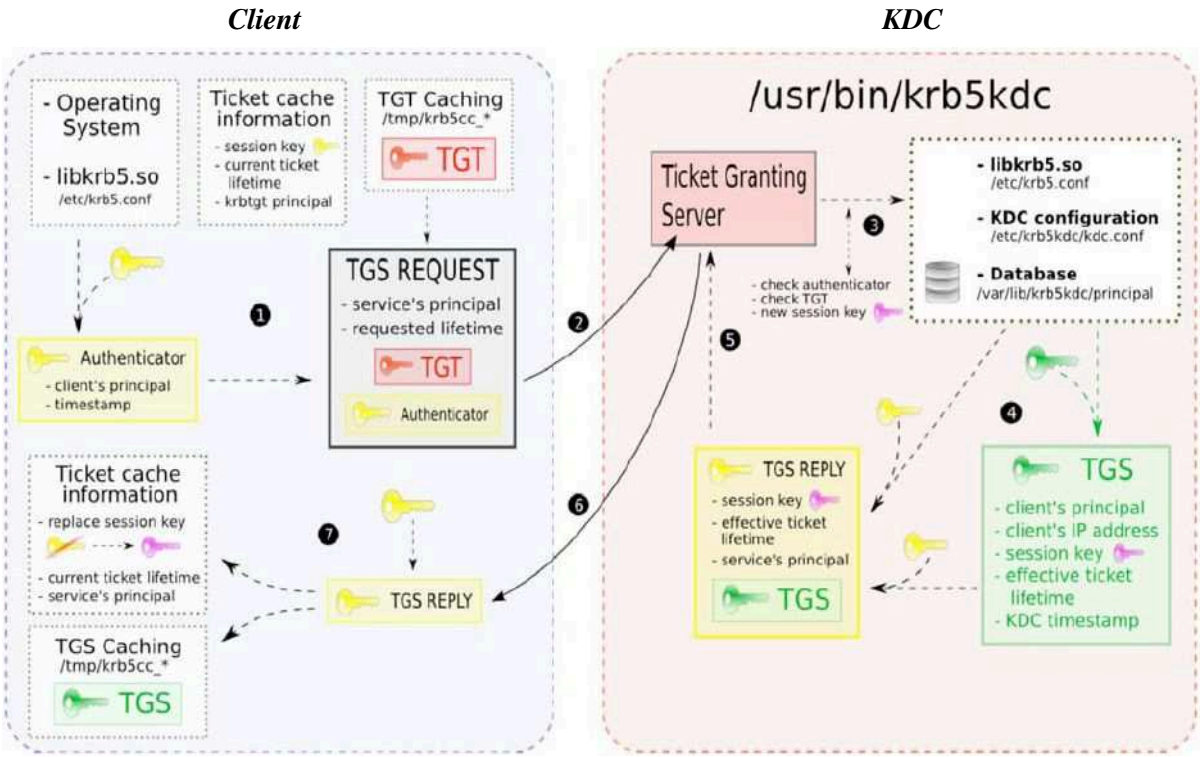
- Kopii nového klíče relace pro službu
- Efektivní timelife lístek
- KDC timestamp
- Principál klienta
- IP adresa klienta

Jakmile klient získá svůj TGS, bude jej používat k ověření sama sebe přímo k požadované službě. Od tohoto kroku závisí do značné míry na službě, která využívá službu Kerberos.

Služba má přístup k tabulce klíčů (Keytab), tedy k souboru, kde je uložen dlouhodobý klíč. Tento klíč umožní službě dešifrovat TGS zprávu odeslané klientem a získat všechny informace potřebné k identifikaci uživatele a vytvořit kontext zabezpečení. Stejně jako na proces TGT, timestamp zakódován v TGS je zde pro zmaření reply útoku.

Tradičně, klíč relace (session key) je použit k podepsání nebo cryptování zpráv mezi klientem a službou. Ten poskytují oba koncové body z důvodu kontroly integrity zasílaných zpráv (pokud zprávy jsou podepsány), a nakonec k vytvoření kontextu zabezpečení, což je tu z důvodu odposlechu.

Kerberos je taktéž kombinovatelný s ostatními bezpečnostními protokoly, jako TLS / SSL nebo IPsec. Hlavním rozdílem je, že jsou založeny na asymetrické kryptografii (RSA). Zatímco Kerberos je postaven na symetrické kryptografii (DES a AES). S tímto spojením se lze často setkat v PKI (Public Key Infrastructure) prostředí. [10]



Obr. 7: Princip obdržení lísku služby [10]

Soubor keytab pak obsahuje výstupy šifrování *aes256-cts-hmac-sha1-96*, *arcfour-hmac*, *des3-cbc-sha*, *des-cbc-crc* a jiné.

Kerberos implicitně pracuje na portech UDP 464 a TCP 749 pro administraci a nastavení hesla (kadmin). Dále pak KDC používá UDP 88 a TCP 88 pro „*kinit*“, tj. příkaz pro žádost o autorizaci od Kerbera. [36]

2.2. Úvod do biometrie

2.2.1. Princip biometrických systémů a jejich optimální nasazení

Jak popisuje kapitola 2.1.3, biometrie je metoda autentizace založená na rozpoznávání jedinečných biologických charakteristikách subjektu. Existuje souhrn výpočetních technik, které dovolují automaticky rozpoznat jakoukoliv osobu na základě jejich fyzických parametrů. Biometrický systém rozpoznávání může pracovat ve dvou různých režimech. Identifikace nebo ověření.

V případě identifikace je proces snažící se zjistit totožnost osoby tím, že zkoumá biometrický vzorek vypočtený z biometrických rysů osoby. Při procesu se sejmou biometrická data neznámého uživatele, která jsou následně porovnána s celou databází. Systém nakonec přiřadí vzorek, který se nejvíce podobá biometrické šabloně. Pro souhlasnou identifikaci musí operace přesáhnout určitou úroveň podobnosti vzorků. Není-li této úrovně dosaženo, vzor je odmítnut.

V případě verifikace je identita osoby požadována apriorně. Vzor je ověřován pouze ve srovnání s individuální šablonou (vzorkem) osoby. Obdobně jako dle identifikace je průběh kontroly systémem stejný, ovšem probíhá podobnost jen mezi vzorem a šablonou. Při kladném výsledku ověření je následně umožněn přístup k zabezpečené oblasti. [15]

Jak již bylo sděleno, identifikace je proces porovnávání (ztotožnění) jednoho ku mnoha vzorkům. Výsledkem je zjištění, která referenční šablona (existuje-li v databázi) odpovídá šabloně vytvořené z nasnímaného vzorku. Verifikace je proces porovnávání jedna ku jedné. Operace jediné šablony vytvořené z nasnímaného biometrického vzorku s jedinou referenční šablonou, patřící prověřované osobě. Cílem je tedy zjistit, zda je prověřovaná osoba opravdu tou osobou za kterou se vydává. Biometrická aplikace potvrzuje nebo vyvrací identitu prověřované osoby. Identifikace je typická pro policejně soudní aplikace, verifikace pak pro bezpečnostně komerční účely. Každé porovnávání má dvě oddělené funkce. Potvrdit, že oprávněná osoba je tou, za kterou se vydává, nebo dokázat, že neoprávněná osoba není tou, za kterou se vydává.

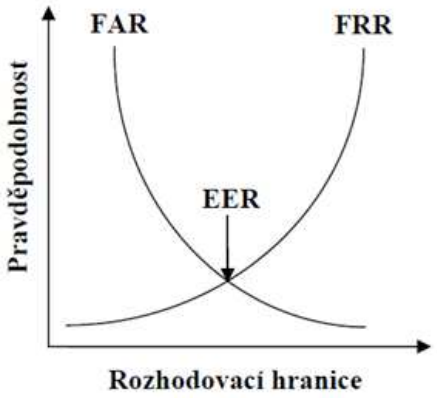
Biometrika jako každý jiný bezpečnostní systém není zcela spolehlivá. Při identifikaci a porovnávání může dojít k dvěma stavům. K chybnému odmítnutí a k chybnému přijetí. Při chybném přijetí je neoprávněná osoba puštěna a identifikována jako oprávněná osoba. Chybné odmítnutí naopak zamítne oprávněného uživatele do systému. V praxi se hodnoty chybného odmítnutí a přijetí většinou neuvádějí v celých číslech, ale v jejich relativních ekvivalentech. To je míra chybného přijetí FAR a míra chybného odmítnutí FRR.

FRR a FAR vyjadřují pravděpodobnost výskytu dané chyby v procentech. Čím nižší je FAR, tím vyšší je FRR a naopak. Hodnota, při které se rovnají FRR a FAR se nazývá EER. [14]

FAR (False Acceptation Rate), neboli koeficient bezpečnosti, vyjadřuje pravděpodobnost, že systém neoprávněně povolí přístup identifikované osobě. Jde o kritickou chybu, jelikož systém přijme osobu, jež za normálních podmínek přístup nemá.

FRR (False Rejection Rate), neboli koeficient "komfortu" vyjadřuje pravděpodobnost, že systém zamítne oprávněné osobě přístup. Tento koeficient nemá vliv na bezpečnost systému, pouze donutí uživatele k opakované identifikaci, což snižuje uživatelské pohodlí.

EER (Equal Error Rate), neboli křížový koeficient udává ideální rozložení koeficientů FAR a FRR. Jeho hodnota určuje oblast, kdy se budou koeficienty FAR a FRR sobě rovnat. [5]



Obr. 8: Rozhodovací hranice [5]

Z Obr. 4 je patrné, že pokud nastavíme vyšší zabezpečení, pak koeficient FRR prudce vzroste, a naopak. Zvyšováním prahové hodnoty vzniká bezpečnější systém (snižuje se pravděpodobnost FAR), ale může docházet ke zvýšení počtu nesprávných odmítnutí (FRR se zvyšuje). To vede k nespokojenosti uživatelů. Naopak snížením prahové hodnoty nebudou uživatelé obtěžováni častými chybnými odmítnutími. Konkrétní nastavení prahové hodnoty závisí na účelu daného systému, vymezení požadavků na zabezpečení systému v závislosti na možném komfortu uživatelů. Z toho vyplývá rozhodnutí, které chyby jsou méně kritické a následuje zvážení rizik plynoucích z těchto chyb.

Důležitou roli hraje počet otisků pro registraci. U každého systému je toto porovnávání omezeno, respektive forma kvality a následného porovnání uloženého otisku. Množství nalezení optimální hodnoty vzorků otisků, neboli biometrických etalonů, což je referenční vzor jeho unikátních biometrických znaků. Tento etalon je běžně brán třikrát, kvůli náhodným jevům, např. nečistotám kůže. Data etalonu jsou zprůměrovány a uloženy v dobré kvalitě. Etalon je uložen buď v tokenu, biometrickém čtecím zařízení, centrální databázi, nebo kombinací těchto metod.

Co se týče vícenásobné autentizace v praxi, je výsledná hodnota FAR dána součinem jednotlivých pravděpodobností FAR a výsledná FRR rovna součtu jednotlivých dílčích pravděpodobností FRR. [14, 15]

2.2.2. Vlastnosti a rozbor otisku

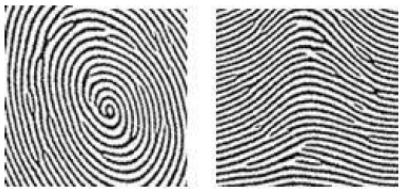
V této práci se jedná o autentizaci pomocí jedinečnosti otisku prstu. Identifikace osob podle otisků prstů patří mezi nejstarší, nejrozšířenější a nejznámější biometrické metody. K identifikaci se využívají papilární linie na konečcích prstů ruky. Papilární linie se nacházejí i na dlaních a chodidlech. Ty se však v současné době nevyužívají pro přístup do systémů, ale pouze ve forenzní sféře. Papilární linie jsou vyvýšené reliéfy, jejichž výška se pohybuje od 0,1 – 0,4 mm a šířka od 0,2 – 0,7 mm. Tyto linie se vzájemně kříží, mění směr, rozvětvují se a spojují, přerušují apod. a vytvářejí tak nejrůznější obrazce, nazývané dermatoglyfy.

Při identifikaci se využívají změny v průběhu papilárních linií, těm pak říkáme markanty. Markant je jakákoliv změna v průběhu papilární linie, která se odlišuje od ostatních. Na tvaru, umístění a vzdálenosti markantů je založeno vyhledávání shodných otisků.

Daktyloskopie zkoumá jakékoli změny v průběhu papilárních linií na konečcích prstů. Identifikace podle otisků prstů je založena na tvaru, umístění a vzdálenosti markantů. Shoda otisku ve forenzní sféře v ČR se potvrdí při shodě alespoň patnácti markantů. U komerčních systémů se pak tato podmínka mění v závislosti na stupni zabezpečení.

Tato technologie se opírá o tři daktyloskopické zákony. Zaprvé neexistují dva jedinci s totožnými papilárními liniemi. Zadruhé obrazce papilárních linií jsou po celý život relativně neměnné a jsou permanentní. Zatřetí je nelze změnit, pouze je-li odstraněna zárodečná vrstva pokožky.

Otisky se mohou získat pomocí statického snímání, kdy se celý prst přitiskne na senzor. Výhodou je intuitivnost. Mezi nevýhody patří nutnost kontaktu se senzorem a možnost zanechání otisku na senzoru. Výhodou jsou menší rozměry a nižší cena. Rovněž na senzoru nezůstává otisk, jelikož ho pohybem uživatel rozmaže. Nevýhodou je pak nutnost správného postupu při pohybu prstu během skenování.

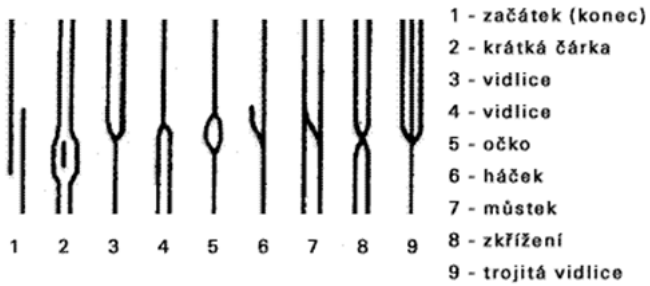


Obr 9: Základní vzory papilárních linií (vír a oblouk) [5]

Při identifikaci se využívá možné rozdělení pomocí základního vzoru papilárních linií:

- *Smyčka* – Kde alespoň jedna papilární linie tvoří smyčku mezi deltou a středem centrální oblasti. Tvoří přibližně 60 procent všech otisků.
- *Vír* – Tvoří ji minimálně dvě delty, přičemž papilární linie vytvářejí oválné, kruhové nebo spirálovité obrazce s jádrem uprostřed. Tvoří přibližně 30% ze všech otisků.
- *Oblouk* – Papilární line zde vytvářejí oblouky, které tvoří přibližně 10% ze všech otisků.

Další rozdělení pak zjišťují zvláštnosti papilárních linií neboli marketů, kde bylo objeveno více jak šedesát druhů markantů. [5, 20]

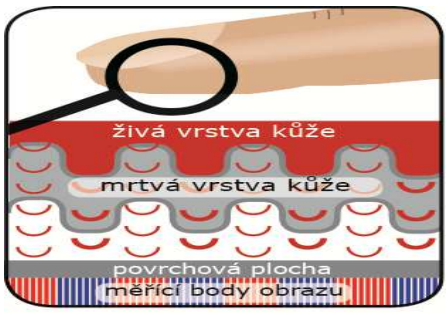


Obr. 10: Markanty

V praxi tato nová technologie nezadržitelně proniká do našeho každodenního života, při otevírání dveří či při vstupování do počítačů identifikujeme pomocí našeho těla. Tato identifikace se realizuje pomocí otisku prstu, dlaně, oční duhovky, obličeje, kartografie žil, tvaru lebky atd. Používání klíčů, magnetických karet, čipů, jmenovek a jiných prostředků ke vstupu do dané místnosti či lokality, přístupu k aplikacím atd. se blíží ke svému konci. Biometrie je jako zrozená technologie, která umožňuje absolutně nezpochybnitelnou identifikaci osob. Biometrie se stala během několika let tím nejmodernějším a nejspolehlivějším způsobem v oblasti kontroly vstupů. [3]

Výhoda biometrie jednoznačně vyplývá z faktu, že ji máme vždy „u sebe“. Nelze tedy předpokládat odcizení, ztrátu, nefukčnost či zapomenutí jako klasické klíče, hardwarovou kartu nebo heslo. Ovšem ruce jsou častým nástrojem v průběhu dne, tedy je tu možnost zranění, pořezání a proto by se mělo se vždy mít na paměti vytvoření alternativ pomocí různých metod.

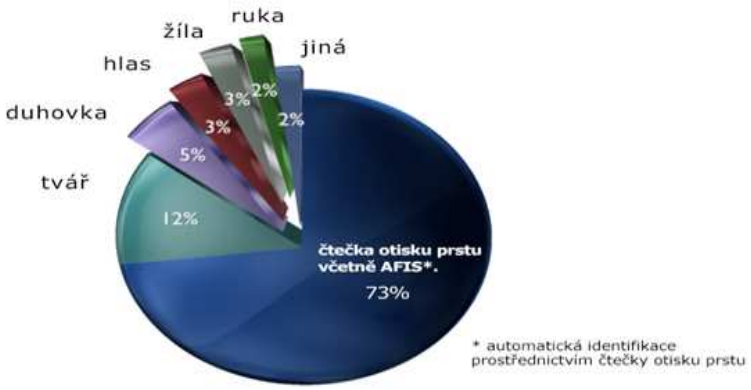
Čtečka při sejmutí vzorku využívá hlavního principu sejmutí otisku díky tažení prstu. Dle různých vlastností povrchů a metod nemusí zůstat na skeneru zbytek otisku prstu, což zabraňuje zneužití nebo falšování.



Obr. 11: Princip čtečky při skenování otisku [2]

Při táhnutí prstu po scanneru se otisknou linie prstu a uloží jako jedinečný kód prstu. Při každém použití se srovnává nově získaný kód prstu s uloženým. Při porovnané shodě otevře impuls dveře, nebo bránu do systému. [2]

Biometrická zařízení na otisk prstu uživatele jsou velmi podporovaná a prodávaná. Může za to pokrok informačních technologií a jejich výroba, což umožňuje širokou dostupnost veřejnosti. Z důvodu malé plochy klesají výrobní náklady a stoupá stabilita. Obr. 9 vyjadřuje podíl trhu s biometrickým řešením. Čtečky na otisky prstů jsou tedy majoritně zastoupeny ve srovnání s dalšími druhy biometrických řešení.



Obr. 12: Podíly cen na trhu biometrických zařízení (International Biometric Group, 2009) [2]

2.2.3. Biometrická přístupová zařízení

Biometrická zařízení pracují ve dvou režimech, a to v registračním a autentizačním. Při přiložení prstu na snímací modul přístroj nasnímá biometrická data od uživatele. Následně v rozpoznávacím modulu dojde k porovnání dat z databáze a v ideálním případě následuje povolení vstupu do systému. Algoritmy pro rozpoznávání otisků prstů se dělí:

1. Podle vzoru – senzor nasnímá otisk prstu. Následně algoritmus zjistí jeho příslušnost k jednomu ze tří základních vzorů a zjišťuje pozici vybraných markantů, popřípadě počet papilárních linií mezi dvěma markanty.
2. Podle podrobností – algoritmus porovnává s etalonem pozici a orientaci jednotlivých markantů v otisku prstu. To klade větší nároky na senzor.

Velkou výhodou je množství zdrojů k zaznamenání referenčních zdrojů (deset prstů). Mezi přednosti také patří již existující velká databáze policie, nízká cena a fakt, že tento postup byl v minulosti dostatečně prozkoumán a ověřen. Nevýhodou pak je možnost obejít systém kopií prstu z odlitku želatiny. Poranění prstu nebo nízká výška reliéfu papilárních linií může mít za následek neschopnost systému identifikovat uživatele. [5]

FRR	< 1,0 [%]
FAR	0,0001–0,00001 [%]
rychlost verifikace	0,2–1 [s]
míra spolehlivosti	vysoká

Obr. 13: Parametry identifikace podle otisků prstů [5]

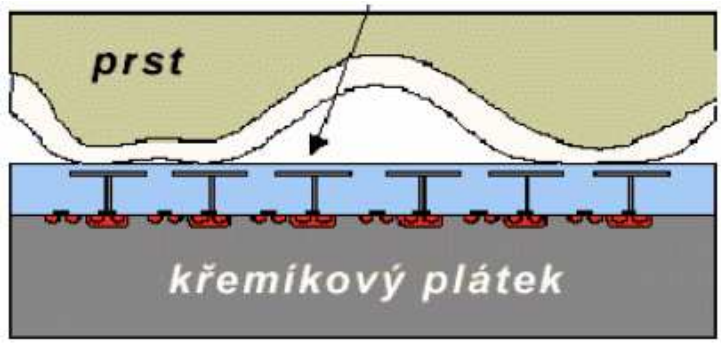
Snímače můžeme rozdělit na kontaktní a bezkontaktní.

- Kontaktní snímače:
  - Optoelektronické
  - Kapacitní
  - Tlakové
  - Teplotní
  - Elektroluminiscenční
  - Elektronické

- Bezkontaktní snímače
  - Ultrazvukové
  - Optické

Mezi Kapacitní snímače se v tomto případě řadí i zde testovaná čtečka Upek eikon. Princip těchto snímačů je založen na měření kapacitního odporu v místě dotyku. Snímač je osázen velkým množstvím mikroelektrod tvořící jednu elektrodu a prst elektrodu druhou. Jedna mikroelektroda tedy tvoří jeden pixel výsledného obrazu. Papilární linie mají tedy větší kapacitní odpor. Tento odpor má vliv na napětí na kondenzátoru, podle kterého je získán obraz papilárních linií. Z principu tohoto snímače plyne, že je extrémně závislý na stavu kůže. Disponuje tedy problémem, kdy při snímání suchých a vlhkých otisk prstů se výrazně mění výsledný kapacitní odpor.

Výhodou je jejich malá velikost a nízká cena. Nevýhodou pak především nízká životnost vlivem elektrostatických výbojů, které vylučuje použití těchto snímačů v některých provozech.
[šmíd]

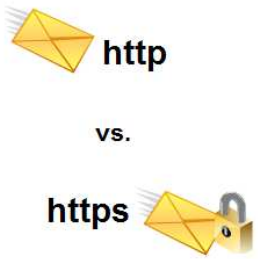


Obr. 14: Schéma kapacitního snímače [35]

2.3. Komunikační protokoly

Každá elektronická komunikace musí probíhat podle definovaného komunikačního protokolu. Protokoly specifikují mnoho základních parametrů. Hardwarové vlastnosti fyzického spojení, procesy navazování a ukončování spojení, detekce a opravy chyb v komunikaci, použité šifrování. Mezi hlavní a nejčastěji používané komunikační protokoly používané na internetu patří skupina protokolu TCP/IP a dále aplikační protokoly, např. HTTP, FTP, POP3, IMAP, SMTP. Většina těchto protokolů existuje i ve verzi, která umožňuje zabezpečené šifrované spojení.

Protokol, který umožňuje spojení mezi webovým serverem a webovým prohlížečem je Hypertext transfer protokol (HTTP), zabezpečená verze HTTPS přenáší data pomocí SSL nebo TLS. Znak „S“ značí vlastnost secure. Standardním portem na straně serveru je 443.



Obr. 15: Internetové protokoly pro webové služby

Protokol HTTPS využívá asymetrické šifrování. Obě strany si před zahájením komunikace vygenerují pár klíčů, vymění si své veřejné klíče a ověří je pomocí otisku veřejného klíče, který je digitálně podepsaný důvěryhodnou certifikační autoritou.

Protokoly SSL (Secure Sockets Layer) a TLS (Transport Layer Security) jsou mezivrstvou vloženou mezi transportní protokol (např. TCP) a aplikační (např. HTTP). Protokol TCP využívá k transportu dat internetem protokol IP, avšak nad tímto protokolem zřizuje spojovanou službu. Musí řešit problémy navázání a ukončení spojení, potvrzování přijatých dat, vyžádání ztracených dat, ale také problémy průchodnosti přenosové cesty.

Protokol TLS je vyšší verzí protokolu SSL, kde oba protokoly však nejsou kompatibilní. Hlavní rozdíl spočívá v rozdílném postupu doplňování datových bloků při symetrickém šifrování. Stěžejním úkolem těchto protokolů je zabezpečit komunikaci šifrováním, autentizovat server oproti klientovi, autentizovat klienta oproti serveru. Klient i server mají jistotu, že komunikace probíhá mezi těmi subjekty, se kterými skutečně chtěli komunikovat. [28, 40]

Navazování spojení probíhá v těchto krocích:

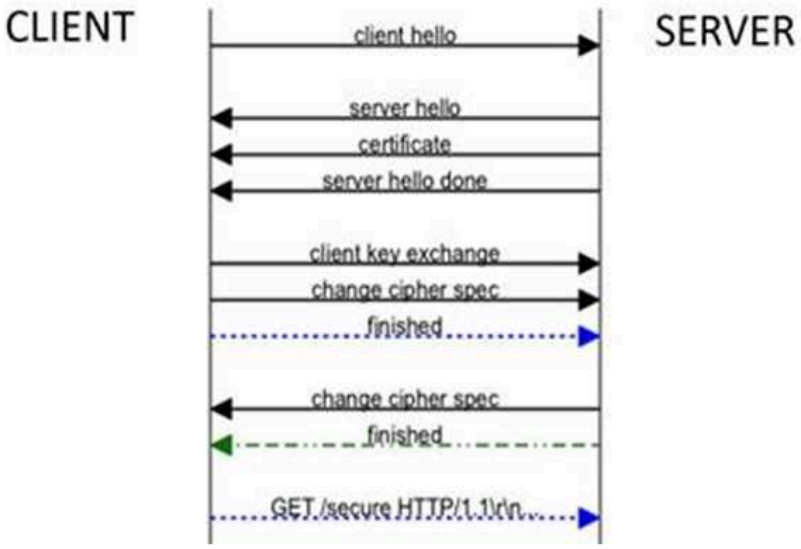
- Klient pošle serveru výzvu k navázání spojení pomocí protokolu SSL a zvolí vhodný šifrovací algoritmus, který budou používat.
- Server odešle klientovi svůj serverový certifikát, který ověří jeho platnost.

- Klient vygeneruje tzv. předběžné sdílené tajemství (premaster secret), které zašifruje veřejným klíčem serveru a odešle ho serveru. V případě, že je požadována i autentizace klienta, je odeslán i jeho certifikát.
- Klient i server použijí předběžné sdílené tajemství k vytvoření tzv. hlavního sdíleného tajemství (master secret), kterým pak šifrují vzájemnou komunikaci. [29]

Pro výměnu klíčů se používají RSA, Diffie-Hellman, DSA nebo Fortezza. Symetrické šifrování následně RC2, RC4, IDEA, DES, 3DES nebo AES. Metody pro jednocestné hašovací funkce MD5 nebo SHA. [wiki ssl]

Z pohledu uživatele jsou samozřejmě tyto kroky transparentní. Na uživateli je pouze zvolení svého platného certifikátu v případě, že server požaduje autentizaci klienta. Další interakce, která je po uživateli požadována, se týká posouzení důvěryhodnosti serveru, respektive jeho certifikátu. Zde může dojít k problému, že webový prohlížeč nemá ve svém úložišti kořenový certifikát autority, která vydala certifikát serveru. Webové prohlížeče v tomto případě prohlásí neznámou certifikační autoritu za nedůvěryhodnou a rozhodnutí o zařazení certifikační autority mezi důvěryhodné ponechají na straně uživatele. [40]

Nejdůležitějším protokolem je Handshake protokol, kterým se komunikující strany dohodnou na použitém šifrovacím algoritmu a klíči. Klient nejprve pošle serveru zprávu ClientHello, obsahující základní informace o použité verzi, dostupných možnostech šifrování a náhodně generovaná data. Server na tuto zprávu klientovi odpoví zasláním zprávy ServerHello, která obsahuje obdobné informace doplněné o certifikát. Poté mu předá aktivitu zprávou ServerHelloDone. Po ověření totožnosti serveru klient může volitelně prokázat svou identitu zasláním vlastního certifikátu, ale vždy musí reagovat zprávou ClientKeyExchange, která zahrnuje náhodná data šifrovaná veřejným klíčem serveru. Po provedení právě popsaných kroků nic nebrání započetí šifrované komunikace. Pokud již bylo spojení mezi klientem a serverem v minulosti vytvořeno, nemusí již probíhat celý proces znovu, ale lze provést obnovení spojení pomocí identifikátoru existující relace. [29, 40]



Obr. 16: Znázornění komunikace mezi serverem a klientem [26]

Certifikát umožňuje objektu se prokázat při elektronické komunikaci a má za úkol svázat totožnost fyzickou s totožností elektronickou daného subjektu. Totožnost je zaručena podpisem certifikační autority. Žádající subjekt musí splňovat řadu kritérií a dodat potřebné dokumenty ověřující jeho totožnost na některou z certifikačních autorit (CA), kde si chce nechat certifikát podepsat. Certifikát si může subjekt vystavit sám na základě dostupných programů a následně si subjekt může certifikát taky sám podepsat (self-signed). Při vlastním podpisu se stává méně důvěryhodný, než kdyby ho podepsala některá ze známých certifikačních autorit. Hlavní částí certifikátu jsou údaje o subjektu, datum vypršení platnosti certifikátu, veřejný a privátní klíč.

Pro generování certifikátu můžeme využít kryptografický nástroj Open SSL. Tento nástroj implementuje Secure Socket Layer (SSL v2 nebo v3) a Transport Layer Security (TLS v1) a další přidružené kryptografické standarty. Tento nástroj pracuje v příkazové řádce a využívá funkce z knihovny krypto, která je napsaná v jazyce C. [33]

3. Praktická část

3.1. Čtečka otisků prstů Upek eikon

Pro ověření uživatele je v diplomové práci využita čtečka otisků prstu Upek eikon. Čtečka se připojuje k rozhraní USB. Výrobce Upek udává, že je vhodná pro operační systém Microsoft Windows, Linux a Mac. Po správné instalaci s pomocí přiloženého CD je v systému Windows vybídnut nový uživatel nejdříve k zaregistrování otisků. Poté je už možno využít přístroj pro biometrickou autentizaci.

Senzor: UPEK swipe sensor TouchStrip® TCS4C
Technologie senzoru: Kapacitní CMOS senzor
Komunikační rozhraní: USB 2.0
Snímací rychlost: Větší než 39 cm/s
Rozměry senzoru: 14 x 4,5 mm
Aktivní snímací plocha: 9,6 x 0,2 mm
Rozlišení otisku :192 x512 pixelů
Pixelový raster: 50µm
Rozlišení: 508 dpi
Certifikáty: CE, UL, FCC, USB 2.0, WHQ

Obr. 17: Čtečka otisků prstů Upek eikon [35]

3.1.1. Instalace čtečky v prostředí Lubuntu 13.04

V operačním systému Linux, konkrétně distribuci Lubuntu 13.04, které je testováno v této práci, je nutné čtečku nejprve nainstalovat. Před instalací čtečky je vhodné vytvořit, popřípadě vlastnit uživatelský účet. Dále je nutné mít v systému potřebné prerekvizity ve formě potřebných balíčků. Nejlépe to uživatel obstará pomocí příkazu v příkazové řádce:

apt-get install libfakekey0 libfakekey-dev libfprint0 libfprint-dev libqca2 libqca2-plugin-ossl libqca2-plugin-gnupg libqca2-dev libpam0g-dev libusb-1.0-0-dev libssl-dev libglib2.0-dev libpolkit-qt-1-dev libmagickcore-dev qt4-qmake g++ libgtk2.0-dev

Instalace pokračuje dále stažením knihovny *libfprint0* například ze zdroje *http://sourceforge.net/projects/fprint/files/*, kde ověřená a otestovaná verze souboru v této práci byla *libfprint-0.1.0-pre2.tar.bz2 (447.5 kB).*

Zabalenou složku nutno extrahovat a pomocí terminálu provést následující příkazy potřebné k instalaci. Popřípadě bude nutné doinstalovat chybějící balíčky, které stanice požaduje.

tar xvfz libfprint-0.1.0-pre2.tar.bz2
cd libfprint-0.1.0-pre2.tar.bz2
./configure --prefix=/usr
make
make install

Pokud je balíček *libfprint0* úspěšně nainstalován, následuje připojení Upek čtečky do portu USB. Vhodné je zkontrolovat výpisem zařízení USB, zdali je čtečka aktivní, a tedy viditelná sběrnicí počítače.

```
klient@klient:~$ lsusb
Bus 002 Device 003: ID 147e:2016 Upek Biometric Touchchip/Touchstrip Fingerprint Sensor
Bus 002 Device 002: ID 80ee:0021
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
klient@klient:~$
```

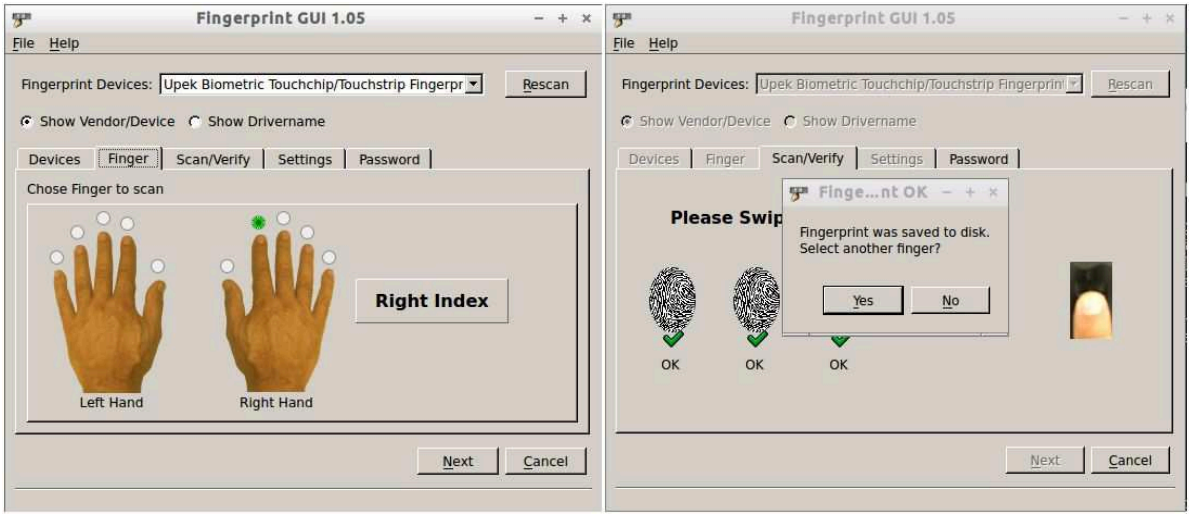
Obr. 18: USB připojená zařízení

Pomocí odkazu např. na adrese *http://darkblue.homeip.net/fingerprint/downloads.php* se uložený, zabalený soubor Fingerprint-gui 1.05 rozbalí a nainstaluje jako dle předešlého postupu instalace knihovny *libfprint0.*

tar xvfz fingerprint-gui-1.05.tar.gz
cd fingerprint-gui-1.05.tar.gz
qmake-qt4
make
make install

Následně se provede instalace konkrétního typu zařízení. Pro senzor Upek slouží příkaz *make install-upek.* Po úspěšné instalaci je vhodné restartovat systém. [13]

Pro linuxový systém se čtečka otisků prstů obsluhuje pomocí programu Fingerprint-gui. Příkazem *fingerprint-gui* je spuštěno z příkazové řádky grafické rozhraní, pomocí kterého se řídí identifikace a manipulace s otisky. Po zobrazení nabídky s detekovaným zařízením Upek Biometric Toughchip Finferptrint Senzor je vhodné postupovat dále k uložení konkrétních otisků. V záložce Finger se zatrhne prst, který se chce uložit. Po minimálně trojnásobně se opakující výzvě k sejmutí prstu rozhraní uloží vytvořený soubor s koncovkou BIR do zvoleného adresáře. Tímto krokem vznikla referenční šablona v binární podobě, která bude sloužit při porovnávání snímaných vzorků.



Obr. 19: Proces sejmutí otisku prstu

Odpovídající soubor s otiskem se uloží do */var/lib/fingerprint-gui/číslo_prstu_zleva.bir*. Na Obrázku je použit ukazováček pravé ruky, tedy označen jako *6.bir*. Dle výpisu pomocí *ls –l* jsou zobrazena práva k souboru, která v tomto případě odpovídají pouze uživateli root, pod kterým byly otisky snímány.

```
root@pavel-virtual-machine:/var/lib/fingerprint-gui/root/libbsapi# ls -l
celkem 4
-rw------- 1 root root 1104 zář 16 20:20 6.bir
```

Obr. 20: Práva k otisku

Záložka Settings definuje cestu, kde je uložen otisk a možnou volbu pro export otisku. Rovněž je možné touto cestou ověřit korektně nakonfigurované PAM služby. Grafické rozhraní umožňuje, jak udává Obr. 20, testování přihlášení k účtu superuživatele root, nebo přihlášení úvodní obrazovce (v případě Lubuntu 13.04 jednající se o prvek lightdm).

3.2.PAM modul

V systému Linux musí být každý soubor či složka v uživatelském vlastnictví. Databáze uživatelů v prostředí distribuce Linux je uložena v textových souborech z důvodu nenáročné manipulace například s programy, či jejich editace. Informace typu OID, GID, zašifrovaná podoba hesla, skutečné jméno uživatele atd. jsou uloženy v */etc/passwd*.

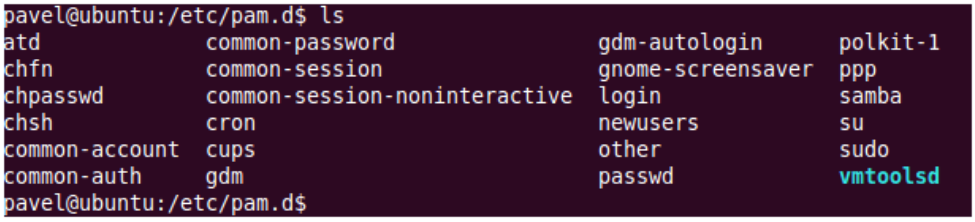
V Linuxu (a Unixu obecně) existuje superuživatel jménem root. Obdobou roota ve Windows je administrátor (správce). Tento správce počítače může dělat naprosto cokoli, takže provádění běžné denní práce pod účtem správce může být velice nebezpečné. Může se napsat nesprávně příkaz a zhroutit si tak systém. Proto je důležité používat jen uživatele s takovými oprávněními, která jsou zrovna pro daný úkol potřeba. V některých případech je třeba použít účet roota, ale většinou (pro klasickou práci na počítači) úplně postačuje běžný uživatel.

Obecně řečeno, *sudo* poskytuje některé vlastnosti, které podněcují určité odlišné pracovní návyky, které mohou mít pozitivní dopad na bezpečnost systému. *Sudo* je běžně používáno na spuštění pouze jediného příkazu, zatímco *su* je běžně užíváno pro otevření konzole a spouštění více příkazů. Přístup sudo redukuje pravděpodobnost zanechání rootovské konzole otevřené donekonečna, a povzbuzuje uživatele minimalizovat používání rootovských práv. Z bezpečnostních důvodů (útoky na heslo pomocí slovníkové metody) byla zavedena tzv. stínová hesla, která jsou uložena v */etc/shadow*. Soubor */etc/passwd* sice stále zůstal čitelný pro všechny uživatele, avšak hesla se přesunula do */etc/shadow*, který dokáží přečíst pouze programy s právy administrátora. Jedním z takových programů je například přihlašovací program. [17, 23]

Implementace PAM modulu do systému zaručuje autorizaci v podobě přečtení hesla, místo toho, aby určité aplikace četly soubor s heslem samy. Modul PAM si pak zvolí metodu autorizace dle nastavení správce systému. Když tedy programy potřebují provést autorizaci uživatele, zavolají funkci, která se nachází v knihovně PAM. Pokud konfigurační aplikace není definovaná, použije se implicitní PAM modul konfigurační soubor. Tento soubor sděluje knihovně, jaké typy ověření uživatele se musí použít. Poté co modul provede kontrolu, vrátí zprávu ověřeno/neověřeno. [17]

PAM modul řeší závislost programů na autentizační databázi, protože funguje jako komunikační vrstva mezi programy a autentizační databází. Programy už autentizační databázi nepoužívají přímo, ale pracují s ní prostřednictvím PAM a nejsou tedy závislé na konkrétním typu autentizační databáze. Pokud se tedy změní autentizační databáze, není nutné přepisovat programy, ale stačí pouze změnit konfiguraci PAM. [35]

PAM objektové soubory jsou uloženy v adresáři */lib/security/*. Moduly samy o sobě nevykonávají žádnou činnost, Aplikace k nim obvykle přistupuje přes dvě vrstvy. První vrstvou je systémová knihovna připojená k programu. Tím se aplikace dostává k dispozici autentizační služby. Druhou nejvýznamnější vrstvou je systémová konfigurace. Zde administrátor určuje, co všechno musí uživatel splnit, aby mu byla služba aplikace poskytnuta. V různých verzích se může lokalizace modulů mírně měnit. Systémová konfigurace, která je rozdělená do souborů v */etc/pam.d* udává, kdy a který modul bude využit pro danou akci. Podobu PAM souborů udává Obr. 19. [21]



Obr. 21: Systémová konfigrace v adresáři /etc/pam.d/

Syntaxe autentizačního modulu je popsána níže. Výraz pro zvolenou metodu ověření je složen ze čtyř typů hodnot. První je tzv. funkční oblast (module-type), druhý sloupec je kontrolní značka (kontrol-flag), třetí položka je samotné jméno PAM modulu (module-path), za kterým mohou následovat další volitelné parametry (arguments).

[module-type] [control-flag] [module-path] [arguments]

Fáze autentizace funkční oblasti pomocí PAM:

- **account**: Kontrola, jestli existuje uživatelský účet, nevypršela platnost hesla, má právo přístupu ke službě.
- **auth**: Samotné ověření identity (kontrola hesla, biometrik, karty apod.).
- **password**: Změna autentizačních mechanismů (např. kontrola síly hesla apod.).
- **session**: Definuje akce, které se provedou před použitím a po skončení použití dané služby (uvítací hlášení, definice limitu na spotřebovaný čas, loggování, apod.).

Mezi kontrolní znaky se řadí tyto atributy:

- **requisite** (povinný): Pokud tento modul selže, PAM okamžitě vrátí výsledek "fail" aplikaci a žádné další moduly se ze zásobníku nevolají.
- **required** (bezpodmínečný): Pokud tento modul selže, PAM okamžitě vrátí výsledek "fail" aplikaci, avšak činnost bude pokračovat voláním dalšího modulu v zásobníku. Celý proces ovšem selže, ať bude následně výsledek jakýkoli.
- **sufficient** (postačující): Pokud modul neproběhne úspěšně, přechází se na další modul v zásobníku. Naopak pokud tento modul uspěje, PAM vrací výraz "pass" aplikaci a žádné další moduly v zásobníku se již nevolají. To vše za předpokladu, že REQUIRED modul neselhal výše v zásobníku.
- **optional** (volitelný): Výsledek "pass" nebo "fail" tohoto modulu je ignorován, což obecně znamená, že modul je volán spíše k provedení nějaké operace, než k účasti rozhodnutí modulu výsledku v zásobníku. Například, pam_keyinit modul se používá jako OPTIONAL modul u sshd, pro vytvoření nové relace klíčů pro nové přihlášení.
- **include** (vložit): Modul zahrnuje všechny řádky daného typu z konfiguračního souboru.
  [18]

Mezi významné a často používané moduly v distribuci Linux se řadí například modul *pam_unix.so*. Jedná se o tradiční UNIXový modul pro autentizaci heslem, který získává ověřovací hesla z */etc/passwd* nebo */etc/shadow*. Modul *pam_deny* je zamykací PAM modul a může být použit k odepření přístupu. Modul *pam_permit* je promiskuitní modul a vždy povoluje přístup. Tento modul je velmi nebezpečný a měl by být používán s extrémní opatrností. Modul *pam_fingerprint-gui* je modul pro autentizaci uživatelů založeném na otiscích prstů.

Volitelné parametry PAM:

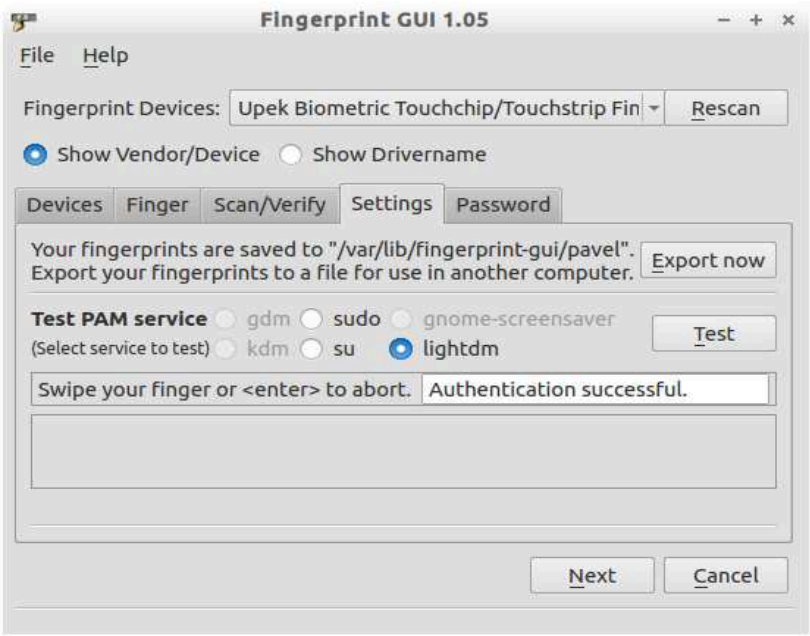
- **nullok:** Výchozí akce tohoto modulu nepovolí uživateli přístup ke službě, pokud heslo má prázdné znaky. Nullok argument přepíše toto výchozí nastavení.
- **use_first_pass:** Modul by neměl vyzvat uživatele k zadání hesla, pokud ho nezískal ze dříve zadaného hesla z předchozího *auth* modulu. Pokud akce selže, pak uživatel nebude ověřen. (Tato možnost je určena pouze pro *auth* a *password* moduly).
- **try_first_pass:** Modul by měl pokusit uživatele ověřit s jeho dříve zadaným heslem z předchozího *auth* modulu. Pokud akce selže, pak je uživatel vyzván k zadání hesla. Tato možnost je určena pouze pro *auth* modul.
- **nullok_secure:** Povoluje uživatelům přístup ke službám, dokonce i když jejich heslo je prázdné, ale jen tak dlouho, pokud je PAM_TTY rovno jedné z hodnot zjištěných v */etc /securetty*.
- **debug:** Zasílá debugovací zprávu do syslogu. Využívá *auth facility*.
- **try_first_identified:** Pokud např. *pam_fingerprint-gui* modul je volán více než jednou v PAM zásobníku a autentizoval uživatele dle jeho otisku v již předchozí autentizaci, pak modul vrací PAM_SUCCESS bezprostředně bez žádosti opětovně o scan otisku. [32]

3.2.1. Nastavení přihlašování pomocí čtečky Upek

Pro nastavení přihlášení do distribuce Lubuntu pomocí čtečky otisku prstů Upek eikon, je nutné využít a editovat PAM. Jedná se konkrétně o soubor *common-auth*, který se stará o přihlášení uživatelů. Lubuntu využívá displej manžera lightdm, který je spuštěn při úvodní obrazovce startu systému. Pro přihlášení je nutné vložit do PAM následující konfiguraci.

auth sufficient pam_fingerprint-gui.so debug

Po uložení */etc/pam.d/common-auth* je případně správnou konfigurace otestovat v grafickém rozhraní Fingerprint GUI dle Obr. 20. Jestliže je po spuštění testovacího tlačítka v menu programu nabídnuta volba k sejmutí otisku prstu, a je úspěšně autentizována, je konfigurace správná. Bez této testovací volby by při nesprávné a nefunkční konfiguraci mohl být i znemožněn přístup do systému natrvalo.



Obr. 22: Testování konfigurace PAM pomocí Fingerprint GUI 1.05

3.3. Návrh SSO implementace

3.3.1. Topologie sítě

Jako autentizační nástroj byl zvolen protokol Kerberos verze 5. Byl vybrán z důvodu jeho open-sourcové distribuce, široké podpory praktických implementací a dalších parametrů popsaných v teoretické části. Zejména podpora jednotného přihlášení a silné autentizace zdobí tento protokol, vyvinutý institutem MIT.

Topologie sítě je navržena na prefixu IP adresy 192.168.183.0/24. Z důvodu požadavků Kerbera je pro veškerou komunikaci stanic využito doménových jmen. Realm sítě je *MYSITE.ORG*. MIT Kerberos je tvořen KDC serverem. KDC se dělí na část AS, která v topologii poskytuje autentizaci hostů, a část TGS zprostředkovávající přístup ke službě. KDC je zde realizován jako samostatný linuxový server, vlastnící doménové jméno *ubuntu.mysite.org*. Pomocí switche, jako aktivního prvku, je připojen k dalším zařízením v síti, jak udává Schéma 1.

Jako server služby na obou webových serverech je v topologii zvolen softwarový server Apache2. Především z důvodu jeho jednolitosti a vhodných předpokladů k testování jednotného přihlášení.

Klientská stanice, která v tomto případě měla adresu 192.168.183.100 ze zvoleného IP rozsahu, vytvářela logické rozhraní mezi klientem, čtečkou otisku prstu, KDC a metodou přístupu ke službě.

Prvek PAM je zde uveden z důvodu koordinace autentikačních mechanizmů, a to dle různých bezpečnostních požadavků. Modul je schopen skloubit jednotlivé různé ověřovací mechanizmy, a to v závislosti na důraz zabezpečovacího systému.

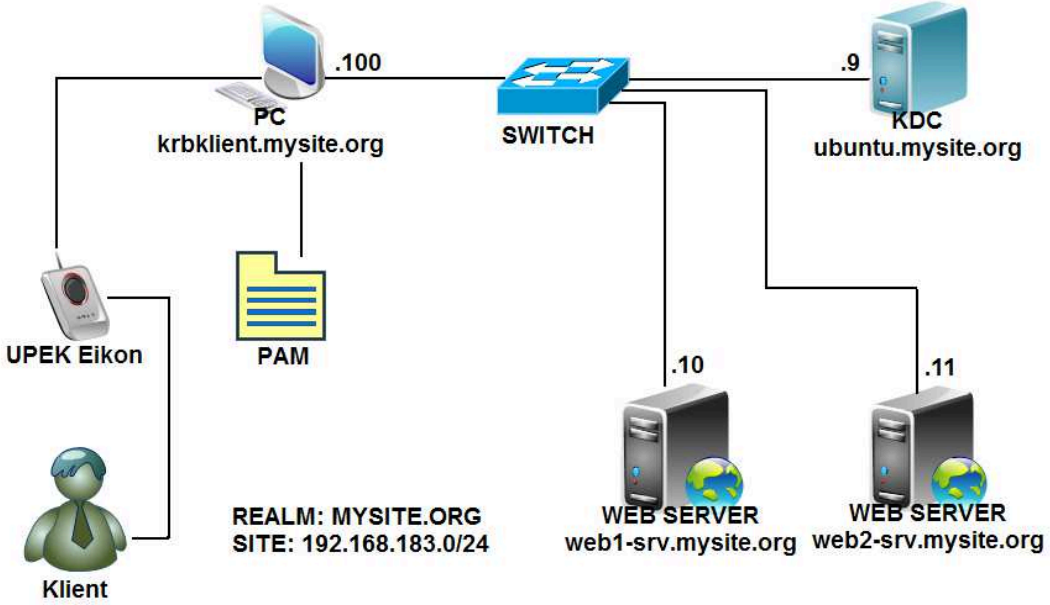


Schéma 1: SSO přihlášení s podporou čtečky otisku prstu

Průběh autentizace by měl být totožný s teoretickým předpokladem funkce Kerbera v5 s tím rozdílem, že dojde předtím k načtení a porovnání otisku prstu uživatele. O volání modulů se postará PAM, který spustí danou ověřovací službu a ta porovná otisk se svou databází uložených vzorků otisků. Uživatel je v ideálním případě ve všech případech úspěšně ověřen a přesměrován na požadovanou webovou službu.

Průběh instalace a nezbytné kroky konfigurace popisuje kapitola 3.4. Detailní popis průběhu autentizace uživatele je uveden v teoretické části v kapitole 2.1.1.

3.3.2. Časová synchronizace

Pro správnou funkčnost a udělení lístku klientovi je nutné mít správně nakonfigurovaný čas i časové pásmo (EDT, CET….) na všech zařízeních v síti, které spolu komunikují na základě protokolu Kerberos v5. Přesným časem je myšlena synchronizace aktuálního času nastaveného na stroji KDC s časem na ostatních stojích. Pro synchronizaci času ze zdroje KDC je zde využito NTP serveru, nainstalovaného na KDC, a to následujícím postupem.

apt-get install ntp
/etc/init.d/ntp stop
ntpdate ubuntu.mysite.org
/etc/init.d/ntp start

Příkazem *date* se v konzoli vypíše hodnota kontrolního času. Při editaci času je důležité pracovat v jednom časovém pásmu z důvodu posunutí hodin. Na ostatních stojích je využito balíčku Crontab, pomocí kterého lze periodicky v závislosti na požadovaném nastavení docílit automatických příkazů a dotazovat se různým službám.

crontab –e

* * * * * */usr/sbin/ntpdate ubuntu*

Tímto příkazem je vložen údaj na každý stoj v síti. Po restartu služby se každou minutu dotazuje příkaz *ntpdate* na hostname *ubuntu*. Tím se synchronuzuje čas všech strojů s aktuálním časem nastaveným na KDC s adresou ubuntu.mysite.org. Výpis nastavení lze zkontrolovat dle *crontab –l.* Příkaz *crontab –r* ukončí službu. [7]

3.4. Instalace protokolu Kerberos

Jak bylo uvedeno, systém Kerberos se skládá ze dvou oddělených částí. Obě se však zpravidla provozují na jednom stroji. První část autentizační služba AS (Autentication Server) se stará o počáteční autentizaci uživatele a vydává mu TGT (Ticket-Granting Ticket). Druhou částí je služba TGS (Ticket-Granting Service), která vydává lístky pro konkrétní služby. [22]

Autentizační mechanizmus je první krok v prostředí protokolu Kerberos, což poskytuje uživateli s TGT přístup k nějaké službě. Z eventuality útoku hrubou silou verze 4 je u Kerberos verze 5 z důvodu bezpečnosti zavedena tvz. možnost pre-autentizace. Princip pre-autentizace spočívá v nutnosti identifikace klienta (požadavek na heslo) ke KDC dřív, než získá TGT. Z toho důvodu musí útočník pokaždé kontaktovat KDC, i když se snaží například o změnu starého hesla.

3.4.1. Konfigurace KDC serveru

Instalace serveru KDC poskytují v distribuci Debian balíčky *krb5-kdc* a *krb5-admin-server*. V navržené síti zatím není nutné použití DNS řešení z důvodu nutné komunikace dle doménových jmen. Upravení souboru */etc/hosts* do tvaru uvedeného níže, se dosáhne totožné funkčnosti. Server, na němž poběží KDC, bude pojmenován *ubuntu.mysite.org*.

```
127.0.0.1          localhost.localdomain   localhost
192.168.183.10     ubuntu.mysite.org       ubuntu
```

Následuje konfigurace knihoven Kerbera. Stěžejní soubor */etc/krb5.conf* najdeme s mírnou obměnou obsahu jak na KDC, tak na straně klienta. Souborem se ovlivňuje chování serverových i klientských programů Kerbera, včetně pomocných nástrojů.

```
# Soubor /etc/krb5.conf na stroji ubuntu.mysite.org
[libdefaults]
 default_realm = MYSITE.ORG
 dns_lookup_realm = false
 dns_lookup_kdc = false

 [realms]
 MYSITE.ORG = {
        kdc =ubuntu.mysite.org:88
        admin_server = ubuntu.mysite.org:749
        default_domain = mysite.org }
```

[domain_realm]
.mysite.org = MYSITE.ORG
mysite.org = MYSITE.ORG

[kdc]
profile = /etc /krb5kdc/kdc.conf

[logging]
default = FILE:/var/log/krb5/krb5libs.log
kdc = FILE:/var/log/krb5/krb5kdc.log
admin_server = FILE:/var/log/krb5/kadmind.log

Konfigurace KDC se nachází v domovském adresáři Kerbera na */etc/krb5kdc/*. Vlastní konfigurace KDC se nalézá v souboru *kdc.conf*. V souboru se upřesňují porty komunikace, databáze realmu, délka platnosti lístku, užité šifry a klíče a lokalizace nezbytných logovacích souborů. Soubor *kadm.acl* definuje přístupová oprávnění k databázi Kerbera.

 # Soubor /etc/krb5kdc/kdc.conf na stroji ubuntu.mysite.org
[kdcdefaults]
kdc_ports = 88
acl_file = /etc/krb5kdc/kadm5.acl
dict_file = /usr/share/dict/words
admin_keytab = /etc/krb5kdc/kadm5.keytab

[realms]
MYSITE.ORG = {
database_name = /etc/krb5kdc/principal
admin_database_name = /etc/krb5kdc/kadm5_adb
admin_database_lockfile = /etc/krb5kdc/kadm5_adb.lock
admin_keytab = FILE:/etc/krb5kdc/kadm5.keytab

acl_file = /etc/krb5kdc/kadm5.acl
dict_file = /usr/share/dict/words
key_stash_file = /etc/krb5kdc/.k5stash
kdc_ports = 88
kadmind_port = 749
max_life = 10h 0m 0s

max_renewable_life = 7d 0h 0m 0s
master_key_type = des3-hmac-sha1
supported_enctypes = des3-hmac-sha1:normal des-cbc-crc:normal
default_principal_flags = +preauth
profile = /etc/krb5.conf
}

Soubor */etc/krb5kdc/kadm5.acl* na stroji *ubuntu.mysite.org* definuje přístupová práva uživatelů. Z té defacto vyplývá, že přihlášený principál *admin* realmu *MYSITE.ORG* má veškeré pravomoce. *Kadm5.acl* má následující podobu:
** /admin@MYSITE.ORG **

Vytvoření databáze Kerbera se dá realizovat spuštěním příkazu *kdb5_util*. Na výzvu je nutno zadat heslo, které bude používáno jako hlavní klíč (master key).
kdb5_util create –s

Následuje vytvoření principálu administrátora krbadmin/admin@FIRMA.LOCAL, pod kterým se bude databáze administrovat. Užije se k tomu příkaz *kadmin.local*. Ten umožňuje spravovat Kerbera pouze lokálně, avšak bez autentizace.
kadmin.local -q "addprinc krbadmin/admin"

Aby start Kerbera KDC byl automatický již při startu systému, slouží k tomu následující konfigurace. Také je vhodné po těchto krocích systém restartovat.
chkconfig krb8-admin-server on
chkconfig krb5-kdc on
/etc/init.d/krb5-admin-server restart
/etc/init.d/krb5-kdc restart

Spuštěním nástroje kadmin, což je administrační nástroj Kerbera, se provede příkazem kadmin s identitou uživatele v tomto případě krbadmin/admin@MYSITE.ORG.
kadmin -p krbadmin/admin

Nyní lze v příkazové řádce kadmin začít administrovat databázi, kontrolovat a vypisovat databázi, a to v podobě v různých forem. Výpis např. obsahu databáze lze příkazem *listprincs* z příkazové řádky kadmina (Obr. 21).

Vytvoření principálu uživatele v příkazovém řádku kadminu je možno vytvořit příkazem *addprinc klient.* V tomto případě není nutné do příkazu vložení názvu realmu. Zda-li je po vytvoření principála klient od Kerbera schopen získat TGT se otestuje příkazem *kinit klient* zadaného z klientské stanice. Možnost získané tikety smazat lze provést příkazem *kdestroy*. [11]

```
root@ubuntu:/home/ubuntu# kadmin.local -p krbadmin/admin
Authenticating as principal krbadmin/admin with password.
kadmin.local:  listprincs
HTTP/web-srv1.mysite.org@MYSITE.ORG
HTTP/web-srv2.mysite.org@MYSITE.ORG
K/M@MYSITE.ORG
kadmin/admin@MYSITE.ORG
kadmin/changepw@MYSITE.ORG
kadmin/ubuntu.mysite.org@MYSITE.ORG
klient@MYSITE.ORG
krbadmin/admin@MYSITE.ORG
krbtgt/MYSITE.ORG@MYSITE.ORG
```

Obr. 23: Výpis principálů obsažených v databázi

3.4.2. Konfigurace klientské stanice

Na straně klienta se pro instalaci Kerbera jedná o balíček krb5-user. Jako na straně KDC je nutné upravit doménová jména v /etc/hosts.

127.0.0.1 localhost.localdomain localhost
192.168.183.10 ubuntu.mysite.org ubuntu

Pro konfiguraci knihoven Kerbera pro klientskou stanici je nutná úprava výchozího souboru, který byl vytvořen při instalaci balíčku v /etc/krb5.conf.

```
# Soubor /etc/krb5.conf na stroji krbklient.mysite.org
[libdefaults]
default_realm = MYSITE.ORG
dns_lookup_realm = false
dns_lookup_kdc = false

[realms]
MYSITE.ORG = {
        kdc =ubuntu.mysite.org:88
        admin_server = ubuntu.mysite.org:749
        default_domain = mysite.org
}
```

[domain_realm]
.mysite.org = MYSITE.ORG
mysite.org = MYSITE.ORG

[kdc]
profile = /etc/kerberos/krb5kdc/kdc.conf

[logging]
default = FILE:/var/log/krb5libs.log

Po zadání příkazu kinit pro získání TGT je ověřena správně nakonfigurovaná činnost systému Kerberos. Obr. 22 zobrazuje získaný TGT v cache paměti, umístěný v /tmp/ na stroji krbklient. Také je zde vidět doba platnosti lístku včetně s principálem služby, který jej vystavil. [19]

```
klient@klient:~$ kinit klient
Password for klient@MYSITE.ORG:
klient@klient:~$ klist
Ticket cache: FILE:/tmp/krb5cc_1000
Default principal: klient@MYSITE.ORG

Valid starting     Expires            Service principal
04/29/14 15:09:59  04/30/14 01:09:59  krbtgt/MYSITE.ORG@MYSITE.ORG
        renew until 04/30/14 15:09:55
klient@klient:~$
```

Obr. 24: Obdržený TGT klienta

3.4.3. Přihlášení uživatele ke stanici za pomocí Kerbera

Kerberos autentizaci klienta ke vstupu do systému stanice zjednává balíček *libpam-krb5*. Po instalaci balíčku je následně přidán modul *pam_krb5.so* do knihoven PAM. Výsledná konfigurace v */etc/pam.d/common-auth* má tuto podobu.

auth sufficient pam_krb5.so minimum_uid = 1000

Nutné je mít pro toto ověření nastavenou statickou IP adresu stroje klientské stanice. Nastavení statické IP adresy lze provést v souboru *etc/networking/interface*. Důležité je restartovat službu networking.

```
#/etc/network/interfaces
auto lo eth1
iface lo inet loopback
iface eth1 inet static
        address 192.168.183.100
        netmask 255.255.255.0
        gateway 0.0.0.0
```

Správnost konfigurace PAM lze ověřit opětovným požadavkem na vstup na klientskou stanici. Po vložení hesla, které je obsaženo v databázi Kerbera, je klientovi umožněn přístup. Komunikace mezi stanicemi je zachycena v podobě výpisů dat zachycených softwarem Wireshark na Obr. 32 na straně 47. Oproti tomu špatné heslo vložené uživatelem vede k neudělení lísku dle Obr. 33 na straně 47.

3.4.4. Konfigurace webového serveru

Serverový software Apache2, který je zde implementován, je nejrozšířenější webový server na světě. K vytvoření webového serveru instaluje Apache do linuxového systému démona httpd. Již bez další nutné konfigurace je pod localhost adresou stanice zobrazena výchozí stránka index.html, která se nachází ve */var/www/*.

Webové servery jsou citlivé na běžné bezpečnostní problémy (ohrožení utajení, integrity a dostupnosti dat), neoprávněný přístup, útok DoS. Cílem je tedy omezit útoky na minimum a umožnit rychlou nápravu obnovení serveru.

V této konfiguraci je využita autentizace pomocí autentikační oblasti Kerbera s možností pre-autentizace, SSO přihlašování, SSL zabezpečovací vrstvy a doplňkově i možnost řídit přístup k serveru z IP adresy nebo domény serveru. [1]

Jako webový server je využit Apache2. Popis potřebných balíčků pro samotnou instalaci a podporu Kerbera je uveden níže. Je možné, že následně bude potřeba doinstalovat nutné balíčky a knihovny dle dispozice linuxové verze.

apt-get install apache2 libapache2-mod-auth-kerb libc6 libcomerr2 libgssapi-krb5-2

Spolupráce webové služby a Kerberos autentizace zprostředkovávají módy Apache2. Konkrétně se jedná o mód *auth_kerb*. Výpis dostupných modulů je zaznamenán na Obr. 24.

```
root@ubuntu:/etc/apache2# cd mods-available/
root@ubuntu:/etc/apache2/mods-available# nano aut
auth_basic.load       authn_dbd.load        authz_dbm.load        authz_user.load
auth_digest.load      authn_dbm.load        authz_default.load    autoindex.conf
auth_kerb.load        authn_default.load    authz_groupfile.load  autoindex.load
authn_alias.load      authn_file.load       authz_host.load
authn_anon.load       authnz_ldap.load      authz_owner.load
root@ubuntu:/etc/apache2/mods-available# nano auth_
```

Obr. 25: Dostupné moduly Apache2

Principál je identita, kterou je Kerberos schopen autentizovat. Principála můžou reprezentovat uživatelé, síťové počítače nebo síťové služby. Principál odpovídající síťové službě se nazývá „service principal“ a jeho vytvoření má následující podobu příkazu:

kadmin -p krbadmin/admin
kadmin.local: addprinc –randkey http/web-srv1.mysite.org@MYSITE.ORG
kadmin.local: addprinc –randkey http/web-srv2.mysite.org@MYSITE.ORG
kadmin.local: ktadd http/web-srv1.mysite.org@MYSITE:ORG
kadmin.local: ktadd http/web-srv2.mysite.org@MYSITE:ORG

Výsledný soubor keytab je složen ze 4 možností šifer daného principálu. Zde se jedná o KVNO 2. Metody zabezpečení jsou definovány v KDC zdrojovém souboru a jedná se o aes256-cts-hmac-sha96, arcfour-hmac, des3-cbc-sha1 a des-cbc-crc jak ukazuje Obr. 25: Pomocí těchto metod kryptografie je porovnáván principál při autentizaci. [25]

```
kadmin.local:  ktadd HTTP/web-srv2.mysite.org@MYSITE.ORG
Entry for principal HTTP/web-srv2.mysite.org@MYSITE.ORG with kvno 2, encryption
type aes256-cts-hmac-sha1-96 added to keytab FILE:/etc/krb5.keytab.
Entry for principal HTTP/web-srv2.mysite.org@MYSITE.ORG with kvno 2, encryption
type arcfour-hmac added to keytab FILE:/etc/krb5.keytab.
Entry for principal HTTP/web-srv2.mysite.org@MYSITE.ORG with kvno 2, encryption
type des3-cbc-sha1 added to keytab FILE:/etc/krb5.keytab.
Entry for principal HTTP/web-srv2.mysite.org@MYSITE.ORG with kvno 2, encryption
type des-cbc-crc added to keytab FILE:/etc/krb5.keytab.
```

Obr. 26: Průběh vytváření kebabu principála služby a zápis do keytabu

Soubor /etc/keytab je nutné pak dopravit zabezpečenou cestou na oba webové servery. Popřípadě bude nezbytné upravit práva k souboru s*cp /etc/krb5.keytab web-srv1@192.168.183.11*:.

Webový server v */etc/apache2/http.conf* se nakonfiguruje do této podoby:

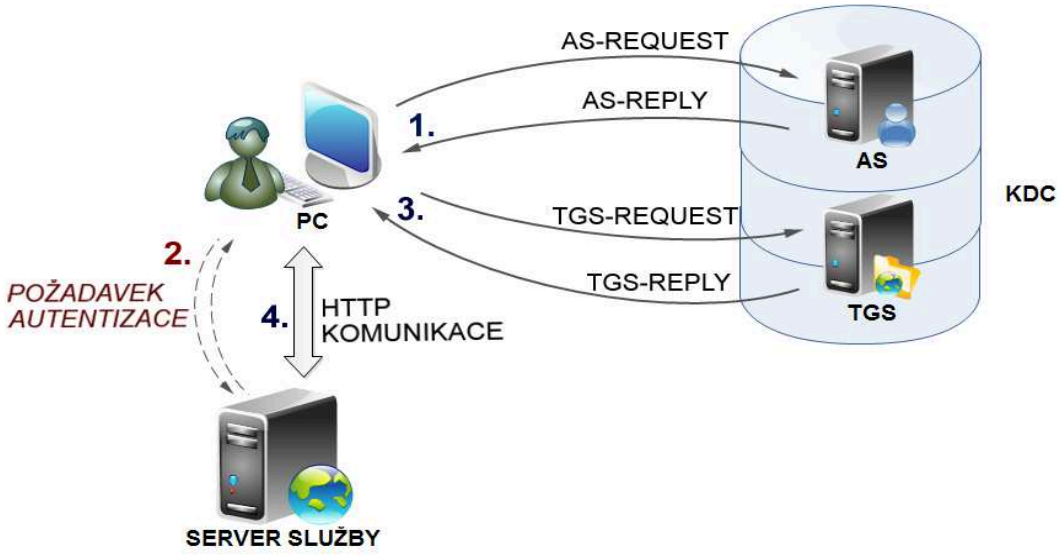
```
LoadModule auth_kerb_module/usr/lib/apache2/modules/mod_auth_kerb.so
<Directory  "/var/www">
        AuthType                Kerberos
        AuthName                "Kerberos Login"
        KrbAuthRealms           MYSITE.ORG
        Krb5Keytab              /home/web-srv1/krb5.keytab
        require                 valid-user
        #order deny, allow
        #allow from localhost .mysite.org
        #deny from all
</Directory>
```

Touto konfigurací souboru se definuje složka, která má být Kerberem autentizována, cesta k uloženému souboru s variací klíčů a také koho se ověření bude týkat. Po restartu Apache2 dle */etc/init.d/apache2 reload* bude webový server aktivní. Autentizace dle PAM požaduje instalaci balíčků:

apt-get install libapache2-webkdc libapache2-webauth krb5-auth-dialog libapache2-mod-auth-pam libapache2-mod-php5

Mod_auth_kerb je modul Apache určen pro ověřování webového serveru pomocí protokolu Kerberos. Modul používá mechanismus Basic auth, který načte dvojici řetězců (username/password) z prohlížeče a zkontroluje jej proti Kerberos serveru. Modul také podporuje metodu „negotiated authentication“ (vyjednávací ověřování), která provádí plnou autentizaci pomocí protokolu Kerberos založené na výměně tiketů, a nepožaduje vkládání hesla do prohlížeče. Aby bylo možné používat vyjednávací metodu, potřebuje od prohlížeče její podporu (Konqueror, IE6.0, Mozilla s rozšířením negotiateauth atd). Mechanismus „negotiated authentication“ může být použit pouze s verzí Kerberos v5. Modul podporuje verze jak Apache 1.x, tak Apache 2.x.

Pokud je používán mechanismus Basic Auth, je využito metody Base64, kterým vznikne datový formát zobrazující binární data pomocí tisknutelných znaků ASCII. Basic modul tedy nedělá žádné zvláštní šifrování jakéhokoliv druhu. U tohoto mechanizmu se snadno údaje mohou převést do čistého textu, proto z tohoto důvodu se s použitím Basic Auth kombinuje s mod_ssl nebo Apache-SSL. Použití SSL šifrování je také doporučeno, i pokud se používá vyjednávací metoda Negotiated. [24]



Obr. 27: Průběh přístupu klienta dle autentizace Kerbera

3.4.5. Využití mechanizmu SSO u webových služeb

Zda-li je konfigurace úspěšná, lze ověřit tím, že pro přístup ke službě se zažádá o konkrétní lístek služby příkazem *kinit –t* HTTP/web-srv1.mysite.org@MYSITE.ORG.

Pro podporu jednorázového přihlášení je potřeba správně nastavený webový prohlížeč. Z webových nejpoužívanějších prohlížečů byly v tomto testovány prohlížeče Firefox, Chromium a Konqueror. Pro první dva uvedené není podpora SSO ve výchozím nastavení. Při vstupu na web-srv1.mysite.org se zobrazí nabídka pro vložení klientského jména a hesla. Po vložení těchto údajů je umožněn přístup na tento server, ovšem nejedná se o metodu jednotného přihlášení. Navíc prohlížeče použijí metod basic, která přenáší citlivé údaje v nešifrované podobě po síti.

192.168.183.12	192.168.183.100	TCP	http > 47889 [ACK] Seq=1 Ack=419 Win=6880 Len=0
192.168.183.12	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.12	KRB5	KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
192.168.183.12	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.12	KRB5	AS-REP
192.168.183.12	192.168.183.10	KRB5	TGS-REQ
192.168.183.10	192.168.183.12	KRB5	TGS-REP
192.168.183.12	192.168.183.100	HTTP	HTTP/1.1 200 OK (text/html)

Obr. 28: Vstup klienta na sever služby bez SSO (Mozilla Firefox)

Prohlížeč Konqueror umožňuje SSO již při výchozím nastavení po instalaci. Konqueror poskytuje metodu negotiate authentication, kterou využívá pro ověření všech sítí v realm doméně. Prohlížeč je tedy připraven k užití, jen za předpokladu, že se v uživatelské konzoli klienta zadá příkaz *kdebug dialog*, který spustí grafickou nabídku. Zaškrtnutím v ní nalezené položky pro webový

prohlížeč konqueror. Poté začne prohlížeč adekvátně pracovat v závislosti na Kerberos autentizaci. Po vstupu na webový server se úvodní stránka zobrazí téměř okamžitě, a to bez zadání jakýkoli údajů. Tomu se tak samozřejmě děje pouze v případě, pokud je klient ověřen AS a má TGT. Proběhla tedy jednorázová autentizace AS části Kerbera a od této chvíle je umožněn klientu přístup ke všem službám uvedených v databázi KDC. Takto se klient prokazuje vůči službě po dobu platnosti lístku, tedy 10 hodin. Výpis lístků uložených na straně klienta je uveden na Obr. 29.

Z obsahu *klist* je vidět, že se klient nejprve autentizuje ke KDC, kde dostane povolení v podobě lístku od krbtgt/MYSITE.ORG, a to v realmu MYSITE.ORG. Klient následně vstoupil k oběma službám v síti, jak dokazují uložené lísky v paměti na stroji u klienta.

192.168.183.100	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.100	KRB5	KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
192.168.183.100	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.100	KRB5	AS-REP
192.168.183.100	192.168.183.12	TCP	35557 > http [ACK] Seq=370 Ack=695 Win=7232 Len=0
192.168.183.100	192.168.183.10	KRB5	TGS-REQ
192.168.183.10	192.168.183.100	KRB5	TGS-REP
192.168.183.100	192.168.183.12	HTTP	GET / HTTP/1.1
192.168.183.12	192.168.183.100	HTTP	HTTP/1.1 200 OK (text/html)

Obr. 29: Průběh SSO Konqueror

```
klient@klient:~$ klist
Ticket cache: FILE:/tmp/krb5cc_1000
Default principal: klient@MYSITE.ORG

Valid starting     Expires            Service principal
04/04/14 21:43:46  04/05/14 07:43:46  krbtgt/MYSITE.ORG@MYSITE.ORG
        renew until 04/05/14 21:43:44
04/04/14 21:47:12  04/05/14 07:43:46  HTTP/web-srv1.mysite.org@MYSITE.ORG
        renew until 04/05/14 21:43:44
04/04/14 21:47:38  04/05/14 07:43:46  HTTP/web-srv2.mysite.org@MYSITE.ORG
        renew until 04/05/14 21:43:44
klient@klient:~$
```

Obr. 30: Výpis lístků na straně klienta

3.4.6. Zabezpečení komunikace webového serveru

Jak je uvedeno výše, modul mod_auth_kerb využívá defaultně metodu basic, která není vhodná z důvodu používání algoritmu BASE64. Oproti tomu metoda Digest, která již neposílá hesla v plain textu po síti ovšem obsahuje další nebezpečí v útoku man-in-the-middle a heslo obsažené na serveru. Také nemusí být řešení plně podporováno prohlížečem. Z toho důvodu je vhodné data požadující silnou autentizaci doplnit a ochránit pomocí zabezpečení komunikace ve formě SSL.

SSL certifikát je způsob šifrování informací na webu a vytvoření lepšího zabezpečeného připojení. Navíc, certifikát může zobrazit identifikační údaje webového serveru pro jeho budoucí návštěvníky. Certifikační autority mohou vydávat ověřené SSL certifikáty o pravosti a podrobnostech o serveru, zatímco certifikát podepsaný sám sebou (self-signed) nemá žádné potvrzení skutečnosti od třetí strany. SSL zprostředkovává šifrování relace asymetrickou kryptografií. Níže je popsána konfigurace pro SSL mód. [1]

sudo a2enmod ssl
sudo service apache2 restart
sudo mkdir /etc/apache2/ssl

Při SSL je výměna dat mezi klientem a serverem šifrovaná. K vytvoření klíčů je potřeba externích nástrojů. Modul mod_ssl přidává systému Apache2 podporu SSL. OpenSSL, který je obvykle zahrnutý u linuxových systémů, poskytuje šifrovací knihovny, nástroje a základní protokoly k tvorbě SSL.

Dle následujícího příkazu vytvoříme vlastní self-signed SSL certifikát. Zde specifikujeme dobu trvání, délku šifry, standard openssl, umístění a pojmenování certifikátu podepsaný sám sebou a definici klíče serveru.

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/apache.key -out /etc/apache2/ssl/apache.crt

Nastavení virtuálních hostů Apache2 je uvedeno níže. Pro SSL je využito portu 443. V souboru je také nutné přepsat aktuální cestu k vygenerovaným souborům.

#nano /etc/apache2/sites-available/default-ssl
<VirtualHost _default_:443>
…
ServerName example.com:443
SSLEngine on
SSLCertificateFile /etc/apache2/ssl/apache.crt
SSLCertificateKeyFile /etc/apache2/ssl/apache.key

Aktivování služby a restart apache:
sudo a2ensite default-ssl
sudo service apache2 reload
[4, 27]

Obr. 31: Obdržený certifikát klienta

3.5. Testování zabezpečení pro různé možnosti autentizace

Jak je známo, o autentizaci se stará PAM. První možností je využívat každou autentizační metodu zvlášť pomocí typu *sufficient*. Autentizace uživatele proběhne v pořádku, pokud alespoň jedna metoda skončí úspěchem.

```
auth    sufficient      pam_fingerprint-gui.so debug
auth    sufficient      pam_krb5.so            minimum_uid = 1000
```

Zdali je z jakéhokoli důvodu vyžadována vícefaktorová autentizace s důrazem na uspění obou těchto metod, bude v *common-auth* následující konfigurace.

```
auth    requisite       pam_krb5.so            minimum_uid = 1000
auth    requisite       pam_fingerprint-gui.so debug
```

Takto zabezpečený přístup je velmi spolehlivý, jelikož napodobit otisk prstu současně se zabezpečením Kerbera je komplikované, ne-li nemožné v omezeném čase prolomit.

Takové zabezpečení je velmi silné, ovšem je stěžejní nejprve konfiguraci ověřit např. v prostředí fingerprit-gui. Mohlo by se stát, že bude vyžadována nekorektní autentizace otiskem, což povede k dostupnosti systému natrvalo.

Co se týče kombinace těchto vícefaktorových autentizací je nevhodnější zvolit Kerberos autentizaci jako *sufficient*, jelikož dotaz na heslo již nabízí standardní linuxový modul *unix.so*. Systém bude zabezpečen a uživatel nebude tolik obtěžován dvojitým ověřením v podobě hesla.

Obr. 32: Úvodní obrazovka Lubuntu 13.04

Ověření pomocí Kerbera je patrné na Obr. 32 a Obr. 33, který zachycuje pomocí analýzy paketů program Wireshark. První zachycuje špatně zadané klientské heslo, tedy zamítnutí vstupu do systému. Po opětovném zadání korektního hesla je vstup umožněn.

192.168.183.100	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.100	KRB5	KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
192.168.183.100	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.100	KRB5	AS-REP

Obr. 33: Přihlášení klienta ke stanici

192.168.183.100	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.100	KRB5	KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
192.168.183.100	192.168.183.10	KRB5	AS-REQ
192.168.183.10	192.168.183.100	KRB5	KRB Error: KRB5KRB_AP_ERR_BAD_INTEGRITY

Obr. 34: Zamítnutí přihlášení klienta ke stanici

3.6. Vlastní testování

Z praktických důvodů byla otestována bezpečnost navrženého systému. Dle teoretických předpokladů je šance na neoprávnění vstup, tedy útočníkem nezařazeným v databázi, do systému značně malá. Pokud se vezme v potaz pravděpodobnost prolomení hesla Kerberos systému a současně pravděpodobnost oklamání čtečky prstu.

Při vlastním testování se vychází z předpokladu, že útočník heslo zná. Jeden atribut (heslo) z nastavené dvoufaktorové autentizace je tedy stoprocentní hodnota FAR. Následně přichází na řadu otázka, jestli je tato znalost pro útočníka dostatečná pro neoprávněný vstup do systému. Hodnoty uvedené v Tabulce 1, vyjadřují dvou procentní pravděpodobnost hodnoty FAR, neboli neoprávněné povolení přístupu identifikované osobě.

Testování bylo provedeno na 13 rodinných příslušnících a známých, kteří neměli uloženy otisky v databázi. Každý zkoušel obejít systém 3 až 4 pokusy, a to snímáním každého prstu. Neoprávněné pokusy se porovnávali s 10 etalony uložených v systému. V tomto případě se nejedná o deset různých klientů, ale o vzorek každého prstu uživatele, a to pro eventuální zvýšení pravděpodobnosti FAR.

	Prst	Hlas	Duhovka	Obličej
Typ	Fyzický	Behaviorální	Fyzický	Fyzický
Metoda	Aktivní	Aktivní	Aktivní	Pasivní
FAR	0,0001%	0,28%	0,00078%	0,10%
FRR	<1%	0,01%	0,00066%	<1%
Reálný FAR	2%	2%-5%	0,94%	1%
Reálný FRR	2%	5%-10%	0,99%	10%

Tabulka 1 – Zjištěné hodnoty FAR a FRR pro různé biometrické typy [5, 32, 34]

Hodnoty FAR a FRR jsou v tabulce zmíněny dvakrát, a to v závislosti na rozsahu, podmínkách a dalších faktorech působících na tyto údaje. Z uvedených hodnot lze vidět rozpětí úspěšností udané v procentech, které odpovídá danému typu biometrického ověření. Tyto reálné údaje budou pro následující testování přijatelnější, jelikož odpovídají jak rozsáhlosti tohoto testování, tak i podmínkám.

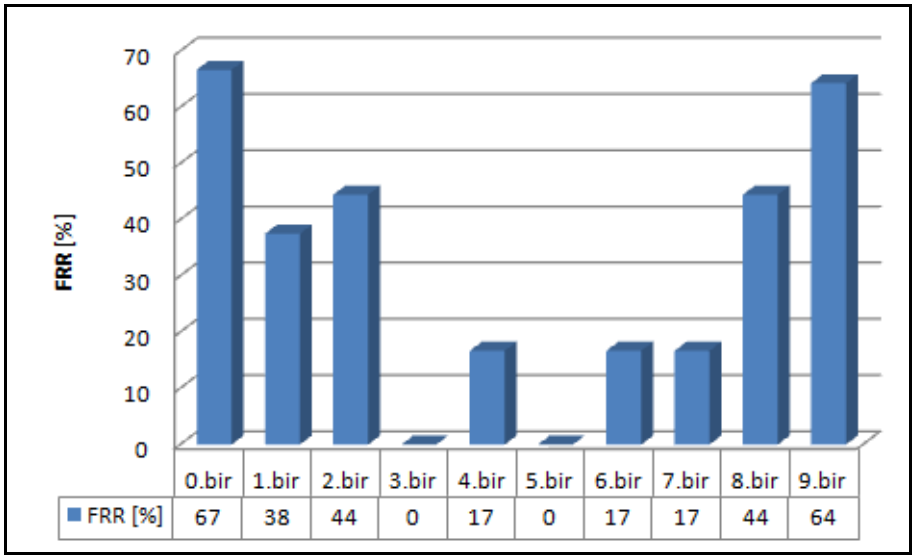
Neoprávnění uživatelé	Uložené vzorky v databázi	Počet pokusů	Přijaté pokusy	FAR [%]
13	10	520	4	0,77

Tabulka 2 – Zjištěné hodnoty FAR

Poté bylo testováno počet zamítnutí FRR, vedoucí zamítnutí oprávněné osobě k přístupu do systému. Testování probíhalo s cílem kolik pokusů scanů je nutné pro 5 přihlášení v rámci jednoho prsu, který je uložen v databázi. Výsledné hodnoty FRR jsou zaznamenány v Tabulce 2 a vykresleny do Grafu 1.

Ruka/Prst	Soubor s etalonem	Počet přístupů	Počet zamítnutí	Počet pokusů	FRR [%]
Pravá/Malíček	9.bir	5	9	14	64
Pravá/Prsteníček	8.bir	5	4	9	44
Pravá/Prostředníček	7.bir	5	1	6	17
Pravá/Ukazováček	6.bir	5	1	6	17
Pravá/Malíček	5.bir	5	0	5	0
Levá/Palec	4.bir	5	1	6	17
Levá/Ukazováček	3.bir	5	0	5	0
Levá/Prostředníček	2.bir	5	4	9	44
Levá/Prsteníček	1.bir	5	3	8	38
Levá/Malíček	0.bir	5	10	15	67
Celkem	-	50	33	83	40

Tabulka 3 – Testování hodnot FRR



Graf 1: Spočtené hodnoty FRR pro každý prst

4. Závěr

Teoretická část diplomové práce je věnována z velké části problematice autentizace jako takové, neboť se jedná o stěžejní pojem v této práci. Jsou zde definovány spolu související pojmy a popsány metody ověření využívající různé technické nebo fyziologické prvky. Popsány jsou nejčastější prvky přihlašování jako heslo, klíč nebo biometrie. Následně jsou tyto metody spojeny do formy vícefaktorové autentizace. S tímto pojmem se váže otázka bezpečnosti a komfortu uživatelů. Není vhodné využívat všech typů dostupných autentizačních systémů či prvků. Naopak je velmi nebezpečné z důvodu určitého komfortu uživatelů využívat pouze jednu zavedenou metodu, například všemi sdílené heslo. O tom pojednává rozhodovací hranice, která stanovuje optimálnost nasazení vícefaktorových přihlašovacích možností. Pokud se jedná o standardní systém, tak křížový koeficient EER by se měl ideálně vyskytovat v polovině charakteristiky, která je dána pravděpodobností průběhů koeficientů FAR a FRR.

Následně je popsán princip metody jednotného přihlášení. Tato problematika s sebou nese náročné teoretické znalosti zasahující do oblasti šifrování, distribuce klíčů, komunikačních protokolů a portů, knihoven a vnitřní struktury operačního systému. Jedná se o mnohdy o robustní, teoreticky a konfiguračně náročně implementovatelné systémy, ovšem které při správné administraci slouží jako silné autentizační mechanizmy. Jako systém vhodný pro použití v této práci byl zvolen MIT Kerberos protokol verze 5. Zejména z důvodu jeho open-source nasazení, silných ověřovacím mechanizmům, absence šíření hesla po síti apod.

V praktické části byla provedena realizace vícefaktorové autentizace za pomocí čtečky otisku prstu Upek eikon a protokolu Kerberos. Prvně byla provedena potřebná instalace a konfigurace Kerbera na všech stanicích v navržené síti obsahující vytvoření kadmina, klienta, databáze, konfiguračních souborů a přístupových práv. Také byla na jednom stroji zprovozněna webová služba v podobě Apache serveru. Po úspěšném přijetí lístku služby na straně klienta byl zprovozněn další webový server. Pro jednotné přihlášení byl použit webový prohlížeč Konqueror, který podporuje tento prvek SSO již ve výchozím stavu po instalaci. Prohlížeče Mozilla Firefox a Chromium byly také testovány a jejich chování je zaznamenáno v kapitole 3.4.5.

K dokonání vícefaktorové autentizace byla zprovozněna čtečka otisků prstů Upek eikon. Veškeré postupy pro provozuschopnou činnost čtečky na systému Linux jsou uvedeny v kapitole 3.1.1. Administrátor nebo i uživatel obsluhuje čtečku pomocí grafického nástroje Fingerprint-GUI. Pomocí toho lze detekovat dané zařízení, sejmout či uložit otisk prstu, či testovat varianty přihlášení.

O přihlášení do systému pomocí čtečky, nebo pomocí Kerbera se stará modul PAM. Tento modul řeší závislost programů, protože funguje jako komunikační vrstva mezi programy a autentizační databází. Ty už nemusí databázi používat přímo, ale pracují s ní prostřednictvím PAM a jsou tedy nezávislé na této databázi. Pak už jen závisí na konkrétní konfiguraci PAM dle nároků na bezpečností požadavky.

Návrh bezpečné autentizace je velmi složitý proces. Důraz je kladen na požadavky technické nebo finanční. Z toho plynou situace, na kterých závisí, jaká a jak velká bezpečnostní rizika budou na systém kladena. Také co se stane v případě ztráty nebo vyzrazení údajů. Na druhou stranu je velmi důležitá případná obtížnost obsluhy a rychlost autentizace, která z toho vyplývá. Autentizace pomocí

sítnice a duhovky je velmi přesná, ovšem vyžaduje finanční náročnost a velkou míru spolupráce uživatele. Své uplatnění má tato metoda v oblasti státních, vojenských, vědeckých nebo i lékařských dat. Použití hardwarových klíčů, či jiných řešení je pohodlné. Ovšem ztráta onoho zařízení je velmi reálná. Pro standardní zabezpečovací systémy je vhodné použít otisk prstu, jakožto důsledek již značné rozšířenosti a cenové dostupnosti čtecích zařízení.

Ideální průběh vícefaktorové autentizace by měl být takový, aby se uživatel autentizoval na daný stroj pomocí svých jedinečných a neměnných vlastností, například ve formě otisku prstu. Následně takto ověřený uživatel již bude autentizován v rámci hesla do různých služeb pomocí Kerbera. Není reálné, aby se tyto metody vykonávaly současně, a to pro každou akci ze strany uživatele. Docházelo by k častému odmítnutí oprávněného uživatele a snížení jeho komfortu. Autentizace pomocí otisku prstu je v dnešní době nejrozšířenější a je založena na jedinečnosti každého jedince, převaze na trhu a nízké ceně. Výhodou je především to, že biometrické prvky nelze zapomenout, odcizit nebo nesprávně umístit.

S přibývajícím nárůstem počtu klientů, nebo rostoucí topologií sítě by realizace Kerbera mohla být z důvodu úložišť kombinována s LDAP databází. Také pokud by nastavením modulu PAM byla Kereberos autentizace stěžejní, je doporučeno obdobně zkonfigurovat a zakomponovat do sítě sekundární KDC server. Při případné nedostupnosti domény KDC by veškerý autentizační systém bez sekundárního KDC selhal.

Vlastní měření testovalo spolehlivost čtečky prstů za předpokladu, že útočník bude znát heslo do systému. Pro tyto účely bylo otestováno 13 rodinných příslušníků a známých, kteří neměli uloženy otisky v databázi. Každý zkoušel obejít systém 3 až 4 pokusy, a to na každý prst. Neoprávněné pokusy se porovnávali s 10 etalony uložených v systému. Z měření vyplynulo, že závisí v prvé řadě na náhodných jevech (tvar prstu, nečistoty, okolní situace). V řadě druhé závisí přece jen na počtu pokusů, kdy se vzrůstajícím počtem přece jen roste šance na špatně algoritmem porovnaný výsledek verifikace. Nicméně z pěti set pokusů došlo na neoprávněný přístup do systému ve čtyřech případech. Tento výsledek dokazuje, že žádný systém není naprosto dokonalý, ovšem s kombinací s autentizací heslem se jedná o systém velmi těžce překonatelný. Naopak testování FRR vykazuje velkou míru odmítnutí oprávněného uživatele. Nejvhodnější prsty z pohledu biometrického ověřování se jeví palec a ukazováček.

Nepřímý výsledek z testování je také ten, že testovaní lidé měli velkou nedůvěru až obavu ze čtecího zařízení. Mnohdy nepochopili jeho snímací princip a svoje pokusy museli nejednou opakovat. Zřídka byla ze strany dotazovaných osob i prosba o toto relativně rychlé testování odmítnuta. Výsledky chování nastávajících uživatelů se také musí vzít v potaz v budoucím nasazení systému.

Závěrem lze konstatovat, že navržená topologie je funkční a lze ji uplatnit v praxi, jelikož se jedná o velmi bezpečnou a levnou vícefaktorouvou autentizaci. Pokud bude nasazení systému optimální a uživatelé budou s funkčností tohoto řešení seznámeni, je toto řešení schopno z pohledu perspektivy uspět.

Literatura

[1] HUNT, Craig. Linux: síťové servery. Praha: SoftPress, c2003, 672 s. ISBN 80-864-9759-3.

[2] EKEY BIOMETRIC SYSTEMS. Co je biometrie? [online]. 2014 [cit. 2014-05-01]. Dostupné z: http://www.ekey.net/co-je-biometrie

[3] Vítejte na internetových stránkách Biometrie s.r.o. [online]. 2014 [cit. 2014-05-05]. Dostupné z: http://www.biometrie.cz/

[4] KABIR, Mohammed J. Apache server 2: Kompletní příručka administrátora. Vyd. 1. Brno: Computer Press, 2004, 724 s. ISBN 80-251-0319-6.

[5] FLÍDR, Jakub. Biometrické autentizační metody. BRNO, 2009. Dostupné z: https://dspace.vutbr.cz/xmlui/bitstream/handle/11012/11697/Biometrické_autentizační_metody.pdf?sequence=1. Bakalářská práce. Vysoké učení technické v Brně. Vedoucí práce Ing. Jiří Sobotka.

[6] ŠČUREK, Radomír. Biometrické metody identifikace osob v bezpečnostní praxi [online]. VŠB TU Ostrava, 2008 [cit. 2014-05-05]. Dostupné z: http://www.biometrickypodpis.cz/PDF/biometricke_metody.pdf

[7] Kerberos: Official Documentation [online]. [cit. 2014-05-01]. Dostupné z: https://help.ubuntu.com/10.04/serverguide/kerberos.html

[8] SCHLENKER, Anna. Behaviorální biometrie pro multifaktorovou autentizaci v biomedicíně. In: [online]. 2012 [cit. 2014-05-01]. Dostupné z: http://www.ejbi.org/img/ejbi/2012/5/Schlenker_cs.pdf

[9] TAKÁCS, ENDRE. *Systémy jednotného přihlášení* [online]. Brno, 2011 [cit. 2014-05-05]. Dostupné z: http://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=42921. Bakalářská práce. Vysoké učení technické v Brně.

[10] MIGEON, Jean-Yves. MIT KERBEROS CONSORTIUM. The MIT Kerberos Administrator's How-to Guide: Protocol, Installation and Single Sign On [online]. 2008 [cit. 2014-05-01]. Dostupné z: http://www.kerberos.org/software/adminkerberos.pdf

[11] Kerberos: ubuntu documentation [online]. 2013 [cit. 2014-05-01]. Dostupné z: https://help.ubuntu.com/community/Kerberos

[12] PŘIBYL, Tomáš. ICT SECURITY. Single-Sign-On a jeho použití v praxi [online]. 2010 [cit. 2014-05-01]. Dostupné z: http://www.ictsecurity.cz/odborne-clanky/single-sign-on-a-jeho-pouziti-v-praxi.html

[13] BIOMETRICS-SOLUTIONS.COM. Upek Eikon Fingerprint Reader on Ubuntu 10.04 [online]. 2010-2013 [cit. 2014-05-01]. Dostupné z: http://www.biometric-solutions.com/devices/index.php?story=upek_eikon-ubuntu

[14] KUBICA, Roman. Biometrie otisku prstu [online]. Brno, 2011 [cit. 2014-05-01]. Dostupné z: https://dspace.vutbr.cz/xmlui/bitstream/handle/11012/2112/roman_kubica_BP.pdf?sequence=1. Bakalářská práce. Vysoké učení technické v Brně.

[15] SYRIS TECHNOLOGY CORP. Technical Document About FAR, FRR and EER: Version 1.0. 2004. Dostupné z: http://ftp.syris.com/SYRIS_ACS_DVD-ROM/UserGuideManual/Reader/SYRDF5-S2MS%20&%20SYRDF6-PMS/About%20FAR_FRR_EER.pdf

[16] JAIN, Anil K, Patrick FLYNN a Arun A ROSS. Handbook of biometrics [online]. New York: Springer, c2008, x, 556 p. [cit. 2014-05-01]. ISBN 978-038-7710-419. Dostupné z: http://books.google.cz/books?id=WfCowMOvpioC&pg=PA8&lpg=PA8&dq=biometric+system+far+frr&source=bl&ots=xoYK5Rs4Kg&sig=Y7lPF3M4Sgk8A8PPVB9bk5IoZsU&hl=cs&sa=X&ei=qUFdU7jPAoTStQa18YCQBA&ved=0CHUQ6AEwCw#v=onepage&q=biometric%20system%20far%20frr&f=false

[17] SHAH, Steve, Wale SOYINKA. Administrace systému Linux: překlad čtvrtého vydání [online]. 1. vyd. Praha: Grada, 2007, 426 s. [cit. 2014-05-01]. ISBN 978-80-247-1694-7. Dostupné z: http://books.google.cz/books?id=8KUzFBDAT6EC&printsec=frontcover&dq=Administrace+syst%C3%A9mu+Linux&hl=cs&sa=X&ei=FjtiU97F4fXsgbIkIHQCg&ved=0CDkQ6AEwAA#v=onepage&q=Administrace%20syst%C3%A9mu%20Linux&f=false

[18] How PAM works [online]. 2009 [cit. 2014-05-01]. Dostupné z: http://www.tuxradar.com/content/how-pam-works

[19] Kerberos V5 System Administrator's Guide [online]. [cit. 2014-05-01]. Dostupné z: http://web.mit.edu/kerberos/krb5-1.10/krb5-1.10.1/doc/krb5-admin.html#Introduction

[20] Obrazce a znaky kůže. [online]. [cit. 2014-05-01]. Dostupné z: http://krimi-spk.sweb.cz/02_exper/expertiz/02a_dakt/02a_kuze.htm

[21] BOBČÍK, Boleslav. ROOT.CZ. PAM - správa autentizačních mechanismů [online]. 2000 [cit. 2014-05-04]. Dostupné z: http://www.root.cz/clanky/pam-sprava-autentizacnich-mechanismu/

[22] VOJTĚCH, Jan. Autentizační systémy [online]. [cit. 2014-05-01]. Dostupné z: http://www.fi.muni.cz/~kas/p090/referaty/2011-jaro/ut/kerberos.html

[23] PAŘÍK, Tadeáš. Root sudo [online]. 2013 [cit. 2014-05-01]. Dostupné z: http://wiki.ubuntu.cz/root_sudo

[24] SOURCEFORGE.NET. Kerberos Module for Apache [online]. [cit. 2014-05-01]. Dostupné z: http://modauthkerb.sourceforge.net/

[25] SHAW, Graham. MICROHOWTO. Create a service principal using MIT Kerberos [online]. 2010–2013 [cit. 2014-05-01]. Dostupné z: http://www.microhowto.info/howto/create_a_service_principal_using_mit_kerberos.html

[26] FOREFRONT TMG TEAM. ISA 2006 / TMG 2010: DISABLE CLIENT-INITIATED SSL RENEGOTIATION, PROTECTING AGAINST DOS ATTACKS AND MALICIOUS DATA INJECTION [online]. 2013 [cit. 2014-05-03]. Dostupné z: http://blogs.technet.com/b/isablog/archive/2013/09/18/isa-2006-tmg-2010-disable-client-initiated-ssl-renegotiation-protecting-against-dos-attacks-and-malicious-data-injection.aspx

[27] SVERDLOV, Etel. How To Create a SSL Certificate on Apache for Ubuntu 12.04 [online]. 2012 [cit. 2014-05-01]. Dostupné z: https://www.digitalocean.com/community/articles/how-to-create-a-ssl-certificate-on-apache-for-ubuntu-12-04

[28] DOSTÁLEK, Libor. Velký průvodce protokoly TCP/IP a systémem DNS. 2. aktualiz. vyd. Praha: Computer Press, 2000, 426 s. ISBN 80-722-6323-4.

[29] Dostálek, L., Vohnoutová, M., Knotek, M. *Velký průvodce infrastrukturou PKI a technologií elektronického podpisu*, 2. aktualizované vydání. Praha: Computer Press, 2009. 542 s. ISBN 978-80-251-2619-6. strana 381-386.

[30] FRAMPTON, Steve. Linux Administration Made Easy: Chapter 6. General System Administration Issues [online]. [cit. 2014-05-01]. Dostupné z: http://tldp.org/LDP/lame/LAME/linux-admin-made-easy/shadow-file-formats.html

[31] Využití vícefaktorové autentizace prostřednictvím kryptografických zařízení v prostředí Open Source Software [online]. 2009 [cit. 2014-05-01]. ISSN 1213-1539. Dostupné z: http://www.elektrorevue.cz/file.php?id=200000409-192781a215

[32] Pam_unix(8) - Linux man page [online]. [cit. 2014-05-04]. Dostupné z: http://linux.die.net/man/8/pam_unix

[33] HLOSTA, Antonín. Systém pro evidenci personální a mzdové agendy. Ostrava, 2009. DIPLOMOVÁ PRÁCE. VŠB – Technická univerzita Ostrava.

[34] BAZEN, Asker M. Fingerprint Recognition: The backgrounds. In: [online]. [cit. 2014-05-01]. Dostupné z: http://www.cmpe.boun.edu.tr/courses/cmpe58Z/spring2010/files/week%2011%20handout%20fingerprint.pdf

[35] ŠMÍD, Petr. Využití biometrie k přihlašování do systému [online]. VŠB – Technická univerzita Ostrava, 2009 [cit. 2013-05-15]. Diplomová práce. Vedoucí práce Ing. Pavel Nevlud.

[36] PETRÁK, Daniel. Autentifikační protokol Kerberos. Vysoká škola ekonomická v Praze, 2010.

[37] JELÍNEK, Martin. IT SYSTEMS. Autentizační tokeny v praxi [online]. 2008 [cit. 2014-05-01]. Dostupné z: http://www.systemonline.cz/it-security/autentizacni-tokeny-v-praxi.htm

[38] Biometrics [online]. 2012 [cit. 2014-05-01]. Dostupné z: http://www.engineersgarage.com/articles/biometrics

[39] ČERMÁK, Miroslav. Autentizace: Jak vybrat vhodnou autentizační metodu? [online]. 2010 [cit. 2014-05-01]. Dostupné z: http://www.cleverandsmart.cz/autentizace-jak-vybrat-vhodnou-autentizacni-metodu/

[40] ČELEDA, Stanislav. Certifikáty a certifikační autority [online]. Českých Budějovicích, 2011 [cit. 2014-05-01]. Dostupné z: http://theses.cz/id/jx5u0r/diplomov_prce.pdf Diplomová práce. Jihočeská univerzita v Českých Budějovicích.

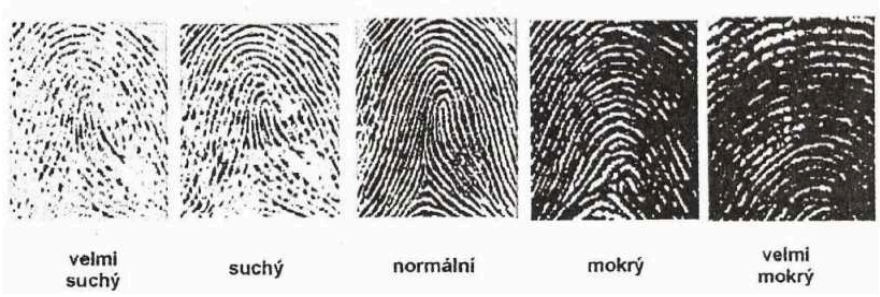
[41] Configure single sign-on authentication on AIX [online]. 2009 [cit. 2014-05-01]. Dostupné z: http://www.ibm.com/developerworks/aix/library/au-configsinglesignon/index.html?ca=drs

[42] NOVOTNÝ, Pavel. Systémy jednotného přihlášení – Single Sign On (SSO) [online]. Praha, duben, 2009 [cit. 2014-05-01]. Dostupné z: http://is.bivs.cz/th/6258/bivs_b/BP_SSO.pdf. Bakalářská práce. Bankovní institut vysoká škola Praha. [43] http://www.fi.muni.cz/~kas/p090/referaty/2011-jaro/ut/kerberos.html

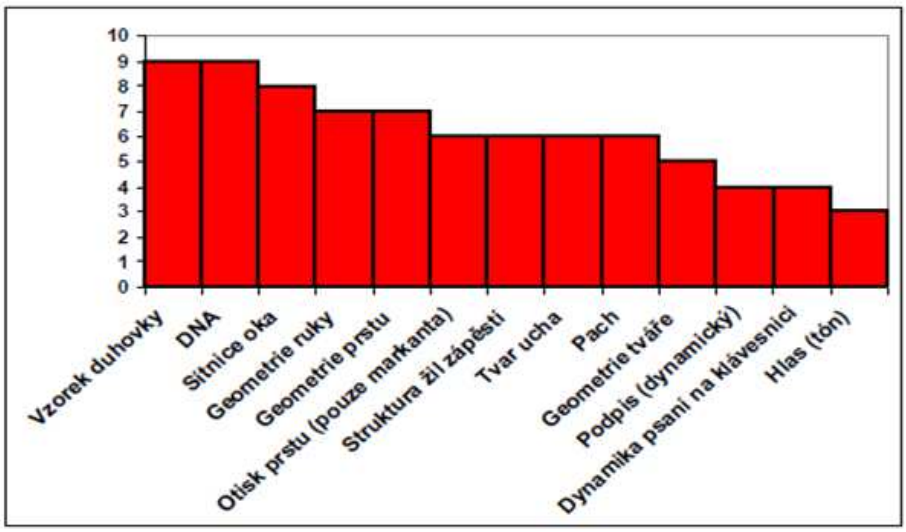
Seznam příloh

Přílohy

Příloha I – Stabilita biometrie

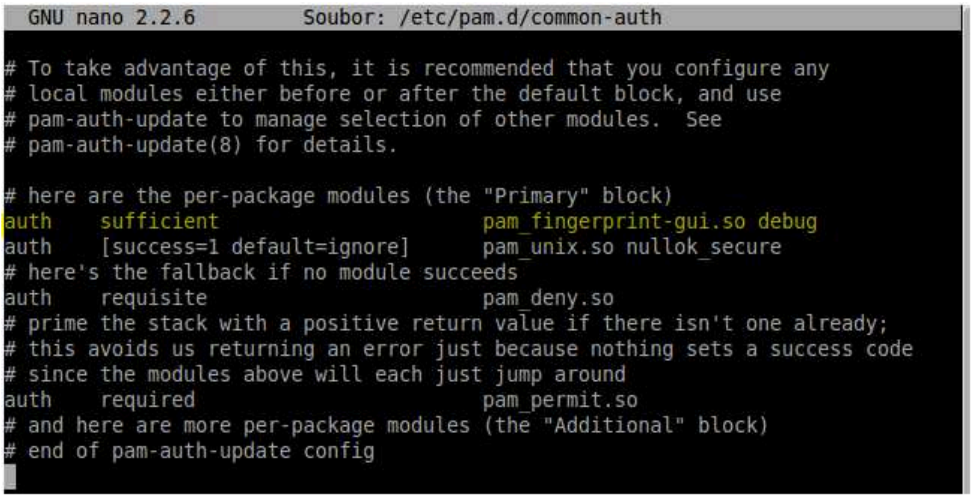


Obr. 1: Vliv nečistot na snímací zařízení [35]

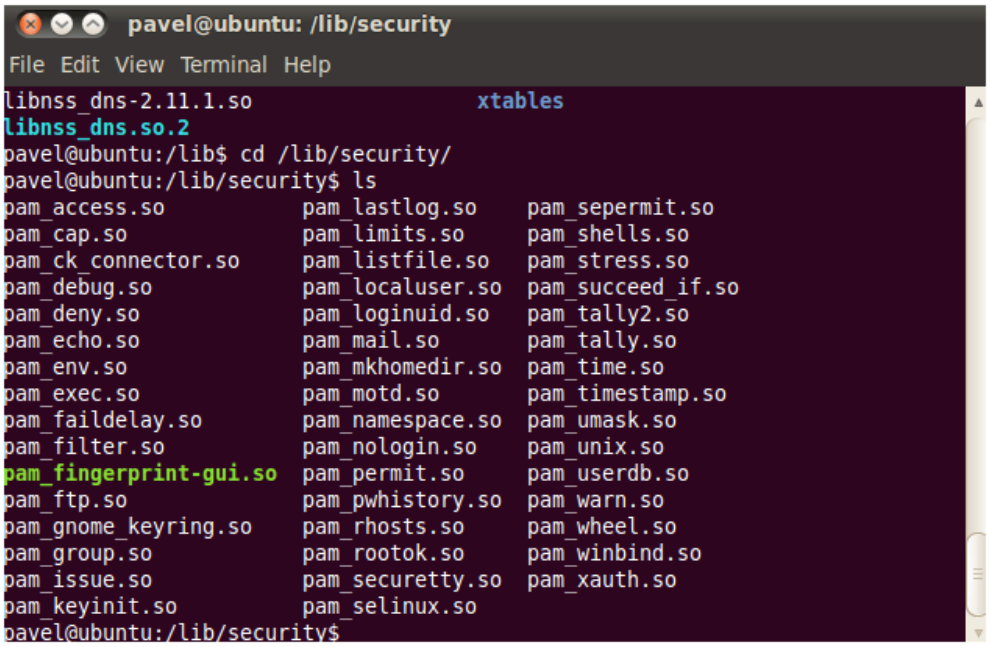


Obr. 2: Stálost biometrických vlastností v průběhu času [6]

Příloha II – Modul PAM v distribuci Lubuntu 13.04

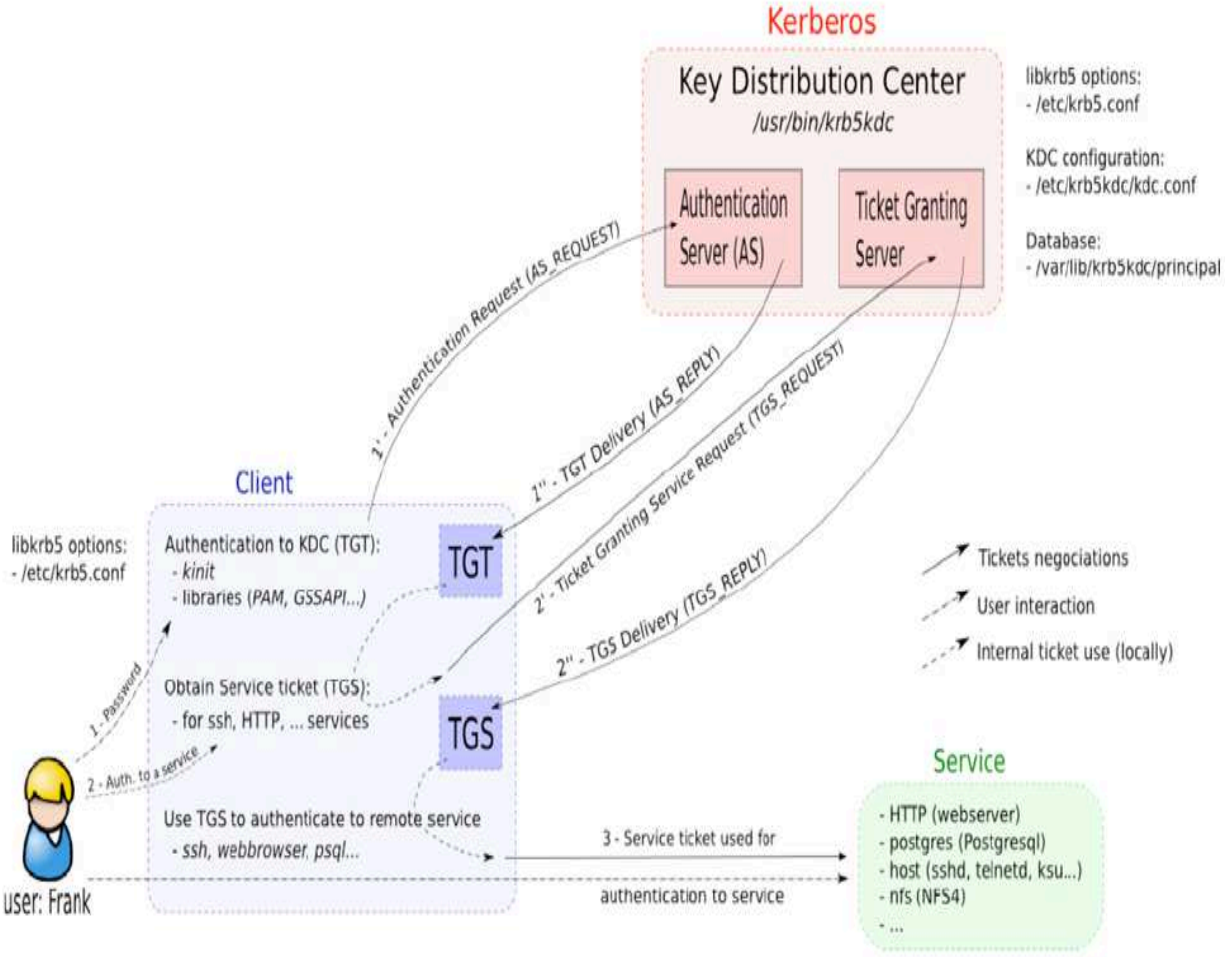


Obr. 1 – Nastavení čtečky pro PAM modul



Obr. 2 – Podoba souboru /lib/security v Lubuntu 13.04

Příloha III – Vyjednávání komunikace MIT Kerberos



Obr. 1: Vyjednávání mezi klientem a KDC

```
 GNU nano 2.2.6          File: /etc/krb5.keytab

^E^B^@^@^@V^@^B^@
MYSITE.ORG^@^DHTTP^@^Spavelweb.mysite.org^@^@^@^AS^WG■^B^@^R^@ ■z■K!■^O^N■■■■Ve$
MYSITE.ORG^@^DHTTP^@^Spavelweb.mysite.org^@^@^@^AS^WG■^B^@^W^@^P■D■^R■<^L■:■■a^$
MYSITE.ORG^@^DHTTP^@^Spavelweb.mysite.org^@^@^@^AS^WG■^B^@^P^@^X■^S■■^M■^S■■4^B$
MYSITE.ORG^@^DHTTP^@^Spavelweb.mysite.org^@^@^@^AS^WG■^B^@^A^@^H■TQ■^Z■■■
```

Obr. 2: Podoba zápisu souboru Keytab